An Evaluation of Saliency and its Limits

Calden Wloka

A dissertation submitted to the Faculty of Graduate Studies
in partial fulfilment of the requirements
for the degree of

Doctor of Philosophy

Graduate Program in
Electrical Engineering and Computer Science
York University
Toronto, Ontario, Canada

July, 2019

Abstract

The field of computational saliency modelling has its origins in psychophysical studies of visual search and low-level attention, but over the years has heavily shifted focus to performance-based model development and benchmarking. This dissertation examines the current state of saliency research from the perspective of its relationship to human visual attention, and presents research along three different but complementary avenues: a critical examination of the metrics used to measure saliency model performance, a software library intended to facilitate the exploration of saliency model applications outside of standard benchmarks, and a novel model of fixation control that extends fixation prediction beyond a static saliency map to an explicit prediction of an ordered sequence of saccades. The examination of metrics provides a more direct window into algorithm spatial bias than competing methods, as well as presents evidence that spatial considerations cannot be completely isolated from stimulus appearance when accounting for human fixation locations. Experimentation over psychophysical stimuli reveals that many of the most recent models, all which achieve high benchmark performance for fixation prediction, fail to identify salient targets in basic feature search, more complex singleton search, and search asymmetries, suggesting an overemphasis on the specific performance benchmarks that are widely used in saliency modelling research and a need for more diverse evaluation. Further experiments are performed to test how different saliency algorithms predict fixations across space and time, finding a consistent spatiotemporal pattern of saliency prediction across almost all tested algorithms. The fixation control model outperforms competing methods at saccade sequence prediction according to a number of trajectory-based metrics, and produces qualitatively more human-like fixation traces than those sampled from static maps. The results of these studies together suggest that the role of saliency should not be viewed in isolation, but rather as a component of a larger visual attention system, and this work provides

a number of tools and techniques that will facilitate further understanding of visual attention.

Acknowledgements

This document owes its existence to many people who have offered advice and support over the course of its development. First and foremost, I would like to thank my supervisor, John Tsotsos, for his continuous guidance. John, your broad well of knowledge and commitment to scientific excellence has been instrumental in my development as a scientist, and I am truly proud to have trained under you.

My family has also given unending support. To my wife, Sarah, you are a vital steadying presence in my life, and you have provided me an invaluable logical sounding board for my ideas and arguments. Your love and advice have been instrumental in allowing me to achieve my goals. To my parents, sisters, nieces, and nephews: your cheers and backdrop of love and support have been appreciated more than I know how to say. To my son, Lou, you may have been a late-comer to this process and not yet quite understand what your daddy works on, but your enthusiastic cheer and games of peekaboo at my defense were bolstering and wonderful. Lastly, I would like to thank my grandfather, who sadly did not get to see this work completed. Granddad, you helped set me on this path, and I hope this work would have made you proud.

Thank you to my labmates and collaborators, who have continuously helped me find new ideas. In particular, I would like to thank Toni Kunić, Iuliia Kotseruba, and Neil Bruce. Toni, your patient troubleshooting and extensive technical knowledge taught me many new skills, and I am immensely grateful to you for keeping my computers running. Iuliia and Neil, you each provided valuable and insightful discussions into the topic of saliency, and I am grateful to both of you for pushing my ideas on the topic to new heights. While the list of additional colleagues is too numerous to mention everyone, I hope you all know how much I have valued your input and critiques.

To my committee, Richard Wildes, Marcus Brubaker, Jacob Beck, Michael Brown, and my

external examiner, Catherine Qi Zhao, I would like to sincerely thank you for your time, ideas, and helpful feedback. You helped push me, and have made this dissertation better and more complete.

Finally, I would like to acknowledge the financial support I received from the National Sciences and Engineering Research Council (NSERC).

Contents

List of Tables

List of Figures

List of Code Listings

List of Abbreviations

AIM	Attention by Information Maximization model
AOI	Area of Interest
API	Application Programming Interface
AUC	Area Under the Curve
AVR	Average Value Ratio
AWS	Adaptive Whitening Saliency model
BBM	Behavioural Bias Model
BMS	Boolean Map Saliency model
CAS	Context-Aware Saliency model
CC	Correlation Coefficient
CIELAB	International Commision on Illumination (CIE) L*a*b* colour space
CLI	Command Line Interface
CNN	Convolutional Neural Network
cG	centered Gaussian
CP	Cognitive Programs
CVS	Covariance Saliency model
DKL	Derrington-Krauskopf-Lennie colour space
DF	DeepFix saliency model
DGI	DeepGaze I saliency model
DGII	DeepGaze II saliency model
DGII_C	DGII saliency model with centre bias
DVA	Dynamic Visual Attention saliency model

DVAP	Deep Visual Attention Prediction saliency model
eDN	ensemble of Deep Networks saliency model
EEG	Electroencephalography
EMD	Earth-Mover's Distance
FES	Fast and Efficient Saliency Model
GAN	Generative Adversarial Network
GBVS	Graph-Based Visual Saliency model
GPU	Graphical Processing Unit
HMM	Hidden Markov Model
ICA	Independent Component Analysis
ICF	Intensity Contrast Features saliency model
ICF_C	ICF saliency model with centre bias
IKN	Itti-Koch-Niebur saliency model
ImSig	Image Signature saliency model
IOR	Inhibition of Return
IQA	Image Quality Assessment
JuddS	Judd Saliency model
KL divergence	Kullback-Leibler divergence
LDS	Learning Discriminitive Subspaces saliency model
LSTM	Long Short Term Memory
MCA	Maximum Central Activation
MIT	Massachusetts Institute of Technology
MS-COCO	Microsoft Common Objects in Context
MSE	Mean Squared Error
Mr-CNN	Multiresolution Convolutional Neural Network saliency model
MVR	Maximum Value Ratio
NSS	Normalized Scanpath Saliency
PCA	Principal Component Analysis
QSS	Quaternion Based Spectral Saliency model

RARE	Rarity-based saliency model
RGB	Red Green Blue colour space
ROC	Receiver Operating Characteristic
RSVP	Rapid Serial Visual Presentation
RT	Response Time
*s*ROC	Shuffled Receiver Operating Characteristic
*sp*ROC	spatial Receiver Operating Characteristic
SalGAN	Saliency using Generative Adversarial Networks model
SALICON	Saliency in Context saliency model
SAM	Saliency Attentive Model
SAR	Separate Activation Regions
SMILER	Saliency Model Implementation Library for Experimental Research
SIM	Similarity Metric
SR	Spectrum Residual saliency model
SSAF	Scale-Space Analysis in the Frequency domain saliency model
SSG	Saccade Sequence Generator model
SSR	Saliency Detection by Self-Resemblance model
STAR	Selective Tuning Attentive Reference model
STAR-FC	STAR Fixation Controller
SUN	Saliency Using Natural Statistics model
VGG	Visual Geometry Group model
VOCUS2	Visual Object detection with a CompUtational attention System 2
WCA	Weighted Central Activation
WTA	Winner Take All
YAML	YAML Ain't Markup Language
YCbCr	Digitized YUV colour
YUV	YUV colour space

Chapter 1

Introduction

Attention is an important aspect of perception and cognition. Despite a long history of studies in attention, understanding and elucidating its properties remains a contentious area of research. Within vision one can find a wide array of attentional effects such as change blindness [13], perceptual switching [14], search efficiency [15], and perceptual priming and inhibition of return [16]. A full account of visual attention, however, remains somewhat elusive, with explanatory proposals ranging from the claim that attention is an emergent phenomenon (*e.g.* the Premotor Theory of Attention [17]) through to that it is a fundamental aspect of sensory processing (*e.g.* the Selective Tuning model [18, 19]).

Given the challenges of accounting for all aspects of visual attention, subfields have emerged that attempt to identify a more tractable scope of investigation. Saliency models form a popular class of computational algorithm aiming to provide a prediction of low-level attention (loosely defined as the capture of attention via exogenous cues). Koch and Ullman's attentional selection architecture [20] first formalized a conceptual model of saliency, influenced by Treisman and Gelade's Feature Integration Theory [21]. Computational versions appeared shortly thereafter, including early formulations by Clark and Ferrier [22], Sandon [23], and Culhane and Tsotsos [24]. Itti *et al.* [25] provided a simple, intuitive model with an accompanying release of implemented code, contributing to the widespread use and interest in saliency modelling. Since then, a large variety of models have been developed ranging from those motivated by the mathematical principles of sparse coding and information theory to those based on machine learning techniques to optimize human fixation prediction without any explicit model of the underlying saliency calculation architecture. See Section 1.2.5 for a more thorough review of existing models. Although most of the theoretical foundations for saliency are rooted in psychophysical results for visual search tasks, saliency algorithms have subsequently been frequently applied as a general purpose form of bottom-up attentional gating (see, for example, [26]), and have been proposed as a useful processing step in a wide range of applications including image compression [27] and robotics [28, 29, 30].

1.1 Motivation and Significance

The field of saliency research has expanded dramatically over the past two decades, including not only the number and nature of saliency models, but also in terms of the applications, performance

metrics, and claims regarding its functional role within vision. With such a rapidly expanding field of research, it seems prudent to examine the practices and assumptions common to saliency modelling.

This work provides three major contributions to the field of saliency research: a critical examination of saliency metrics resulting in a novel performance metric explicitly designed to deal with spatial bias, a unified software design structure intended to make it easier to compare and test saliency algorithms, and the development of an explicit model for saccadic control that extends models of low-level attention beyond a static saliency map.

1.2 Background

1.2.1 Attention, Fixation, and Saliency: Disentangling Terms

Before diving into the details of a set of computational models, it is important to ensure that the philosophical underpinnings of what those models are attempting to accomplish are well-founded. First and foremost to accomplish this task it is necessary to disentangle a number of terms that are too frequently only loosely or implicitly defined, leading to a conflation of disparate topics or a mismatch in interpretations between fields. It should be noted that many of these terms are quite difficult to precisely and explicitly define, as they are often contextual, species-specific, and subject to individual variation. Nevertheless, this section will attempt to provide as clear a definition as possible for a number of terms of interest, noting how the terms will be used within this document with mention of common forms of misuse. Although attention is certainly not unique to the domain of visual processing, this section will specifically be discussing the use of terms as they relate to visual attention.

1.2.1.1 Attention and Attending

Attention is a notoriously difficult term to define, while at the same time it is an intrinsic and vital part of human cognition (as well as for many other species, but the focus here is on human vision). Because of the tight coupling of attention to our routine sensory processing experiences, most people generally have an intuitive sense of what attention is, as well as the sense that everyone understands it and it therefore does not need to be explicitly defined. However, as Tsotsos [19]

argues in comprehensive detail, there is no one single definition of attention, and it is rather best understood as a broad class of mechanisms for controlling and tuning the processing of information. The act of *attending* is the bringing of one or more of these information processing mechanisms to bear on a particular focus.

Although there may be specific situations in which it is possible to attend to more than one stimulus at the same time, referred to as *divided attention*, there are a number of computational constraints that limit the efficacy of such an approach [19], and neurophysiological evidence suggests that directing attention to multiple items simultaneously degrades representation [31]. Unless otherwise specified, therefore, this document will assume that at any given time there is a single locus of attention.

1.2.1.2 Fixation and Selection

Fixation is a term that is frequently conflated with selection in the context of visually attending to a target. *Fixation* refers to the act of maintaining the eye in a relatively steady position in order to gather visual information [32]. A *fixation location* is the point in an image or scene upon which the eye is centered during a fixation period. In humans, this point corresponds to the centre of the fovea, the portion of the retina with the greatest visual acuity [33, 34].

Selection refers to the act of attending to a specific spatial region of the visual field. Given the anisotropy in visual acuity across the retina, it is very common for selection to coincide with an eye movement to bring the selected target into a more optimal view for information gathering. When selection coincides with an eye movement in this manner it is said to be an *overt* allocation of attention. When one attends to a selected location without any corresponding movement of the eyes, in contrast, it is referred to as a *covert* allocation of attention, as there is no external manifestation of this attentional focus. However, researchers are not always consistent with the amplitude threshold used to register a saccade, which can at times lead to difficulties in categorizing attentional allocation as either covert or overt (see the discussion of microsaccades below).

Covert attention has been described as early as Helmholtz's seminal work on physiological optics [35], in which he viewed brief flashes of stimuli consisting of a field of letters and was able to shift which letters were consciously detected by him without moving his eyes. Subsequent research has shown that in monkeys saccades and attention can be disassociated such that instructions

dictating the direction of a saccade may be covertly observed [36], but that saccade target locations nevertheless do appear to have facilitated attention even when subjects are specifically instructed to attend to another location [37]. It is this latter result that is used to justify the use of fixation as a proxy for selection. However, Hunt and Kingstone [38] subsequently argued that when both tasks are speeded (as they were in [37]), this confounds the results, and when only one task is speeded a clear dissociation may be made between selection and saccade target location.

As mentioned above, a further potential confound that ought to be addressed is the magnitude of eye movement size. Even during periods of fixation, the eye is not truly stable, but rather continues to make a variety of very small movements (referred to as *fixational movements*), without which the visual percept of an image fades away [39]. Fixational movements are frequently classified into three categories: microsaccades, ocular drift, and tremor [40]. Rucci and Poletti [41] provide a fascinating recent review of the role of fixational eye movements in visual processing, but also make the point that, given the rapidity and small magnitude of these movements, correct detection and classification is often rather difficult. For this reason, eye tracking research will frequently discard *all* eye movements below a certain threshold of magnitude, whether those movements are saccadic in nature or not. This may cause unnecessary confusion when differentiating between covert and overt attention, as some experimental results that have classically been interpreted as covert are nevertheless accompanied by microsaccadic eye movements [42]. Though these microsaccidic movements may not foveate the target, they nevertheless do trigger the neuronal modulations in processing that have been identified as commensurate with saccades and overt allocations of attention. It is therefore important to keep in mind that though there is a strong link between fixation and selection, the phenomenon referred to by each term is not synonymous, and eye tracking data alone may not reveal the full story of spatial selection.

1.2.1.3 High-Level, Low-Level, Bottom-up, and Top-down

When discussing different information processing steps and approaches in a hierarchical network, there are two pairs of terms that are used frequently but which are rarely explicitly defined: low-level and high-level, and bottom-up and top-down. It is unfortunate, but the two pairs are often conflated, with high-level and top-down used interchangably, and likewise bottom-up and low-level (*e.g.* [4, 43, 44, 45]). This casual mixing of terms causes unnecessary confusion within the literature,

and should be avoided.

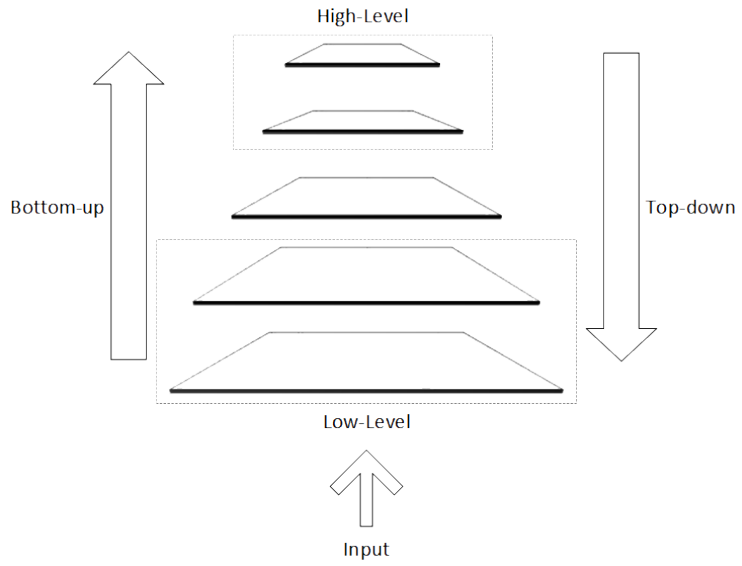


Figure 1.1: A graphical depiction of the terms *low-level*, *high-level*, *bottom-up*, and *top-down* as they pertain to a hierarchical network

Low-level and *high-level* are used to refer to the *level of abstraction* within a processing hierarchy for which specified features are encoded or processing steps take place (represented as dashed boxes in Figure 1.1). They are loose terms that do not precisely localize the item in question, but rather provide a qualitative assessment of whether the item occurs earlier or later, respectively, with respect to the feedforward flow of information within a hierarchical network. Felleman and van Essen's seminal work [46] explored this hierarchical flow of information in the primate visual cortex, determining processing level such that each area must be placed above all areas from which it receives ascending connections or sends descending connections, and likewise must be below all areas from which it receives descending connections or sends ascending connections.

Bottom-up and *top-down*, by contrast, specify the *direction* of information travel within a network (represented as arrows on either side of the network in Figure 1.1). Bottom-up processing corresponds to a feedforward pass of the network; information travels from the input layer of the hierarchy toward the uppermost layers. Bottom-up attentional effects predominantly correspond to exogenous cues (cues originating from external stimuli). Top-down processing, in turn, refers to steps that involve information flowing from higher to lower levels of abstraction. Top-down

modulation may occur independent of any feedfoward passage of information; it can, for example, be set by internal motivation, prior knowledge, or task demands (effects frequently referred to as endogenous, or originating internally). Alternatively, top-down modulation may occur as a form of recurrent feedback that refines earlier input (*e.g.* [47]) or as an attentional process to aid in important perceptual tasks like feature binding (see [48] for an extensive discussion of this topic).

Thus, while features such as faces ([4, 43]), semantic tags ([45]), or depth ([44]), may be considered to be high-level features (particularly when compared to the set of basic features utilized by Itti *et al.* [25]: orientation, colour, and luminance), in these works they are nevertheless still being calculated and used in a strictly feedforward network, meaning that they are decidedly not an example of top-down attention. Top-down modulation need not even occur over high-level features; Navalpakkam and Itti [49], for example, propose a model of top-down modulation of low-level features that aims to maximize the signal-to-noise ratio of the target against its background.

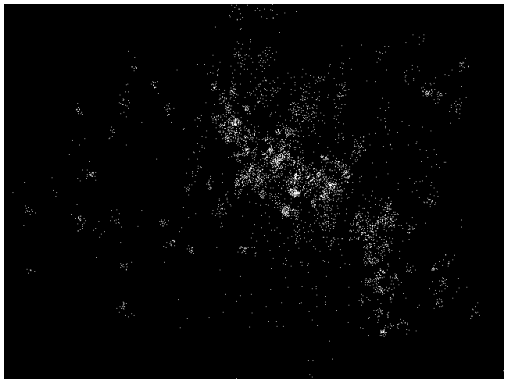
1.2.1.4 Salience and the Saliency Map

As mentioned at the start of this chapter, the term *salience* as it is used in computer vision today has its roots in the work of Koch and Ullman [20] and Treisman and Gelade [21]. It specifically refers to the stimulus-driven attentional pull of a visual element [25, 50, 51, 52]. This is a contextual property, as stimulus salience is relative to its surroundings and may sometimes be modulated in unexpected ways by the competition for selection [53]; what is salient in one context may cease to be salient in another.

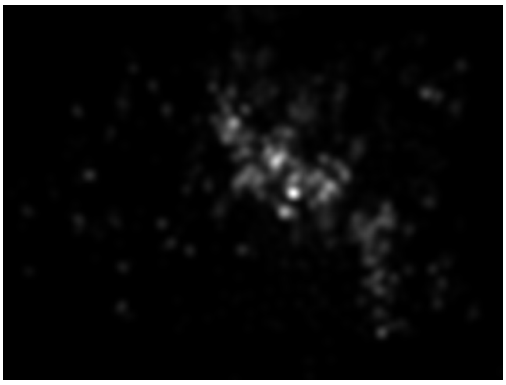
Because of its contextual and somewhat subjective nature (as some subjects may find an item more salient than others), the term salience has often been conflated with other related terms such as *relevance* (*e.g.* [54, 55, 56, 57]), or within contexts of an imposed task (*e.g.* [58]). As Fecteau and Munoz [59] point out, these uses are not consistent with the original use of the term salience, and that salience should be restricted in use to the bottom-up processing of items, and propose instead the more general term *priority* to refer to the fusion of stimulus-driven bottom-up attentional pull with top-down task modulation. Unfortunately, the literature continues to be inconsistent in its use of the term salience and what information is represented by a saliency map. This dissertation will strive to restrict usage of the term salience (and its corresponding saliency map) to the original usage, and will specifically note when that usage deviates from the original concept. However, it

should be noted that a number of models that are referred to as "top-down" saliency models in actuality refer to models that include high-level features (*e.g.* [4, 43, 44, 45]) but are nevertheless strictly feedforward in nature without a task modulation component; such models will therefore be referred to as saliency models without any additional caveats.

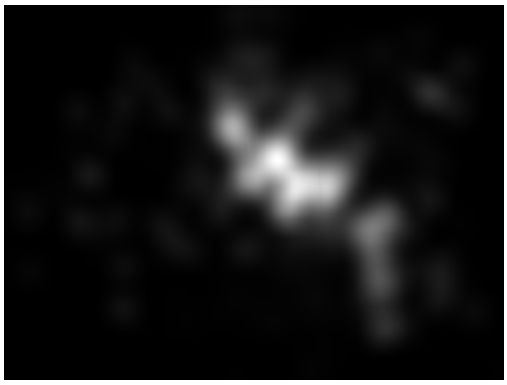
(a) Example image (size 640 × 480 pixels) from the ImgSal dataset [60].



(b) Human fixation points corresponding to the image in (a).



(c) Points in (b) convolved with a Gaussian kernel 20 pixels wide with sigma equal to five pixels.



(d) Points in (b) convolved with a Gaussian kernel 60 pixels wide with sigma equal to 15 pixels.

Figure 1.2: An example showing an image (a) and ground-truth human fixation data (b) along with two smoothed ground truth maps ((c) and (d)) using different kernel specifications.

In the computer vision literature, a *saliency map* is a representation of the distribution of saliency over a visual field. It is typically provided as the output of a saliency model in the form of a static two dimensional array with saliency values assigned to each position in the array representing the salience of the corresponding pixel location in the original input image. However, it should also be noted that this concept is sometimes extended to refer to other forms of fuzzy prediction over a two dimensional spatial field. For example, Simonyan *et al.* [61] use the term

saliency map to refer to the magnitude of pixel contribution to the decision of a neural network to assign a given class label to a particular image. Although there is a link to the usual use of the term "saliency map" in the sense of assigning pixel-wise importance, or in a metaphorical sense trying to highlight which pixels most grab a network's attention when it is looking for a specific object class, it is not a true saliency map in that it does not try to represent task-free exogeneous pull for attention. Many works also refer to an "empirical saliency map" [62], "ground-truth saliency map" [63], or a related term to refer to the transformation of a binary map of human fixation data into a smoother, less sparse map via convolution with a Gaussian kernel (for an example, see Figure 1.2). This is done in order to represent human fixational data in a form that is more amenable to distributional forms of quantitative evaluation for saliency model output (see Section 2.2 for a review of saliency metrics). However, it should be pointed out that these maps are an artificial construction; they do not necessarily correspond to any neuronal representation of conspicuity, and therefore it is something of a misnomer to refer to them as saliency maps at all. Moreover, the parameters used to construct them can have a direct effect on the evaluation of computed saliency maps [62] as well as any qualitative comparisons (note the different appearance of Figures 1.2c and 1.2d), which means it is important to keep in mind at all times that a given empirical map is solely a convenient transformation of human fixation data.

1.2.2 The Role of Saliency

Given the general emphasis in saliency research on human fixation data for performance evaluation, it is often implicitly assumed that the role of a saliency algorithm in vision is to serve as a guide for saccade control (see, for example, [64]). However, within the larger topic of general visual attention, the concept of a saliency algorithm has also frequently been applied as a form of first-order attentional gating to reduce the volume of incoming visual data that must be processed, largely drawing on the original concentration of Koch and Ullman [20] on single image visual processing without eye movements. Keeping in mind the definitions presented in Section 1.2.1, the conceptual requirements and constraints required by the different roles that saliency might play are explored here.

1.2.2.1 Saliency as Early Visual Gating

Early examples of saliency as an early mechanism for the spatial focusing of attention can be found that predate the Itti-Koch-Niebur (IKN) model [25]. Sha'asua and Ullman [65] propose a series of local computations based on edge structure and curvature, the output of which will lead to the preferential subsequent processing of salient regions. Olshausen *et al.* [66] proposed a similar mechanism for early visual detection of target objects based on a pre-attentive saliency map that dynamically adjusts cortical weights to allow subequent visual processing of a target, effectively suppressing the rest of the visual field. Even after the IKN model marked a shift in research focus to computational modelling and performance evaluation using human fixation data, a number of efforts continued to focus on the cognitive role of saliency maps as an early attentional gate. Examples of such work can be found in Li [67], who proposes that V1 cells tuned to features such as orientation and colour create a saliency map at the level of the primary visual cortex that promotes activity at more salient spatial locations, and Navalpakkam *et al.* [68], who take as given a pre-attentive saliency map as a component of scene processing and understanding.

More recent efforts have focused on how to balance such an early-stage stimulus-based attentional gating function with endogenous attentional allocation (based on elements such as semantic instructions, prior experience, or training). Zhang *et al.* [69] demonstrate with monkey recordings the ability of salient non-attended stimuli to disrupt spatial attentional enhancement based on earlier training. Buschman and Kastner [70] propose an attentional model in which attention can be controlled in one of two ways, either through early capture by inherently salient stimuli, or by task guidance. Although the exact balance between task and exogenous capture is not clearly elucidated, in the absence of task guidance it is proposed that attention will be allocated in rank order of salience across the visual scene.

However, none of these previous efforts have actually addressed whether early gating truly is a general attribute of human attention. In fact, the exact nature of an attentional bottleneck has been an ongoing question in the attention literature for many years, and an early bottleneck is not universally agreed upon [19]. Of particular note are tasks for which there is not time to move the eyes, and thus the only attentional shifting possible must necessarily be covert. In these situations if saliency is truly a general purpose early gating system, then stimulus saliency should nevertheless

impact performance.

A number of psychophysical experimental results have demonstrated the ability for human subjects to rapidly categorize visual stimuli [71, 72, 73]. Potter's seminal early work [71] established that subjects shown a rapid serial visual presentation (RSVP) of a sequence of images (with only short periods of exposure for each image ranging from 113ms-333ms) could correctly identify the presence of a target image (for example, "Was there a picture of a boat?") with surprisingly high accuracy. Subsequent work [72] identified that this detection ability degraded as the duration of stimulus exposure was reduced, but nevertheless remained significantly above chance even for exposure times as short as 13ms. Thorpe *et al.* [73] used EEG recordings to identify the evoked reaction potential (ERP) timing for images displayed for 20ms when subjects were asked to identify if the images contained an animal or not. They found that although the image was only shown for 20ms, the time to differentiate the two conditions occurred after 150ms of elapsed time. Collectively, these studies provide strong evidence for the ability of a human observer to perform rapid image classification based solely on a single feedforward pass through the visual system, without the benefit of eye movements over the image.

Although these experiments do not directly answer the question of whether or not humans utilize early gating, human RSVP performance nevertheless provides a useful domain with which to examine the generality of saliency as an early attentional gating mechanism. If there truly exists a pre-attentive saliency map that provides a general-purpose reduction in the volume of visual data to be processed, then human performance would be predicated upon the salience of the target stimulus within its image. After all, there is no time for continual exploration of the image, so if the target is to be detected it must immediately fall within the regions of interest that make it past the saliency gate. Investigations using the same datasets as Potter *et al.* [72] and Thorpe *et al.* [73] find that the tested saliency models do not adequately provide a meaningful prediction of the target location [74, 75]. Although it might be argued that the saliency algorithms tested were simply the wrong ones, and that is why they failed to sufficiently identify the relevant object as most salient, it is important to note that the most relevant visual component is contextual. Most vitally, in a number of the trials conducted by Potter *et al.* [72] the target image was specified after the sequence was shown, meaning that any saliency algorithm operating over the incoming visual stimuli would have to be independent of task-based guidance for a specific target. Thus, it

is unclear if any saliency algorithm exists that is capable of accurately gating the incoming visual stimuli to highlight the appropriate target at levels sufficient to achieve human task performance without the global knowledge of a full pass through the visual hierarchy [76, 77].

1.2.2.2 Measuring Salience and Saliency Map Performance

Much of the challenge in discussing saliency rests with the internal nature of its neural representation and lack of a non-contextual, non-individualistic measure or calculation. Indeed, this is not a unique challenge to saliency, but is rather a broader challenge across much of the research into visual attention. Nevertheless, as saliency models were developed and implemented, there naturally arose the question of how to demonstrate and evaluate the efficacy of these models. The earliest evaluations tended to be through a mixture of approaches, including demonstrating the model's ability to find a singleton target in a visual search array, qualitative judgement of a rank order traversal over an image, and comparing simulated search times in a natural image [25, 64].

Over time, however, as the number and complexity of available models grew, as well as the computational resources available for evaluation, the field heavily shifted toward the task of fixation prediction over natural images as the primary method of saliency model evaluation (see Chapter 2 for an extensive discussion of saliency metrics for fixation prediction). Initially these comparisons were qualified with recognition of the assumptions and limitations inherent in such a comparison (*e.g.* [2]), but this acknowledgment of the potential limitations and drawbacks to basing performance on fixation prediction was largely dropped in subsequent efforts (*e.g.* [78]), and eventually human fixations over natural images became the explicit target to which saliency representations were trained (*e.g.* [4]) and the basis for largescale algorithm comparison and benchmarking [79].

Although human fixation prediction has become one of the primary applications of saliency algorithms and a standard comparitor for algorithm performance evaluation, it is important to continually acknowledge its limitations. Fixations are not synonymous with selection, but rather only encompass overt shifts in attention. Furthermore, saccadic accuracy is not always perfect, or may be placed at a medial location between two salient points rather than directly at either target [80], meaning that fixational data is at best an imperfect proxy for overt attentional shifts. This by no means is intended to suggest that fixation prediction is not a valuable or useful domain over which saliency models might operate, but rather to point out that it should not be viewed as

the singular goal of saliency representation or sole arbiter of saliency map accuracy. To this end, Chapter 4 explores some additional evaluation domains based on psychophysical stimuli.

1.2.2.3 Saliency and Objectness

An enduring problem in visual computing is automatic object detection. The combination of both widespread interest in automated object detection methods and the computational challenges of abstract object representation prompted an early convergence with saliency map research. In many ways the use of saliency as part of an object detector is a special case of early visual gating, in which the focus of the saliency representation is explicitly shifted from finding that which is most conspicuous to finding potential objects in the scene. Shokoufandeh *et al.* [81] provide an early example of this strategy with a coarse-to-fine algorithmic representation of objects that uses a representation of saliency at the coarsest level. However, their work was still largely focused on the goal of object detection with saliency applied as a contributing component; later research efforts inverted the direction of development and focused on the concept of proto-object cues to guiding or improving the representation of saliency.

Influenced by the work of Bar [82] positing top-down modulation to facilitate object detection and recognition via rapid scene context detection, the work of Torralba [83] and Oliva *et al.* [84] represents early efforts proposing a move away from strictly low-level feature directed models to a model driven by higher level scene considerations. Although not directly attempting to detect objects, global scene context is used to prime the presence or absence of objects of interest. This global context is integrated with a traditional feature-based saliency map, preferentially allocating saliency to regions of the image that are likely to contain objects of interest. The approach of processing global image statistics to provide a rough measure of scene context was, along with Oliva and Torralba's work on scene categorization based on its spectral properties [85], an early application to computer vision of the psychological concept of scene gist [86]. This spawned an extensive branch of computer vision research (see, for example, [87, 88, 89]), but was largely superceded within saliency research by more focused local detection of objects in preference to contextual cues to object presence.

Walther and Koch [90] extended the original IKN model to approximate the extent of a target region based on a connected-components-like region grower based on the point of maximum saliency

(the winner-take-all (WTA) pixel) found in the saliency map. Their model allows for region growth based on a heterogeneous contribution of features, as the decision of pixel inclusion is based on the maximum saliency contribution taken across the different feature channels. Ultimately, though, their model includes the fundamental assumption that it is a cohesive object that will win this pixel-wise saliency competition, and their final results are still based on the pixel-wise feature-based saliency map of the IKN model rather than on a principled search for object-like properties.

Object-based saliency began to formalize as a distinct branch of research with Einhäuser *et al.*'s controversial claim that image objects were better predictors of human fixation targets than feature-based saliency [91]. This prompted a schism in saliency research, with one branch of saliency research continuing to focus on pixel-wise maps largely based on feature responses (although some such models do incorporate specific object detectors such as faces; see Section 1.2.5 for more details on specific algorithms) while others focused explicitly on the detection of proto-objects as a driver for saliency. Alexe *et al.* [92] designed a highly influential formalism for the latter approach, defining objectness as being the degree to which a scene element exhibits the following three characteristics:

1. A closed boundary in space
2. Different appearance from the surroundings
3. Uniqueness within the image

Based on these principles, candidate object windows are found in the image, and can either be converted into a traditional saliency map based on window concentration and confidence, or used as location priors to more traditional object detection algorithms. This original paper has since been extended in numerous ways (see in particular [93] for an example that further develops the connection between objectness and saliency).

It should be noted that this formulation implicitly assumes a priori that objects are the most conspicuous scene elements, an assumption that may be problematic. One specific issue arises with cluttered images; with many objects it is unclear how a salient object detector ought to behave. A scene element forming an object is not sufficient to make that element conspicuous when there are many other objects that are also competing for attention. Figure 1.3a shows a highly cluttered scene in which no single object carries clear semantic priority over the other scene elements. However, if

a salient object detector were simply to highlight all object elements it would end up labelling the majority of the scene as salient, which is a rather unsatisfactory result. In a similar vein, Figure 1.3b displays the issue of multiple objects in a highly different scenario. In this street scene there are many cars, all of which carry a degree of semantic interest, as well as a number of business signs (which by their nature are designed to be eye-catching). Should the algorithm preferentially select the vehicles that are more centrally located in the image as they are more photographically prominent, or should it instead highlight the vehicle on the far right as it is the only vehicle in the scene that is not parked? Do the individual buildings themselves count as objects to compete for a viewer's attention, or should a salient object detector ignore them as part of the background? Although the objectness characteristics defined by Alexe *et al.* may well contribute to conspicuity, it is an overly strong assumption that objectness is the preeminent factor in determining the allocation of attention.

(a) Cluttered objects **(b)** A street with many objects

Figure 1.3: Examples of cluttered images taken from the MIT1003 human eye tracking dataset [4]. The figure on the left shows an example of a cluttered image in which none of the objects is likely to inherently be of strong interest in a standard free-viewing task, whereas the figure on the right shows a street scene with many objects all of which are likely competing for the attention of a human observer.

There has been other pushback against the notion of objectness as the root of saliency and eye fixations. Borji *et al.* [94] argue that Einhäuser *et al.*'s original finding was driven by the particular ways in which performance was measured, and more robust measures of performance no longer come to the same conclusion. This highlights the vital importance of taking into account the design and purpose of the metric being used (see Chapter 2 for further discussion of performance evaluation),

but may also suggest that there is not a strong binary dichotomy between object-based saliency and feature-based saliency. It is entirely plausible that there is a balance between the two that shifts through time (as more time allows for more semantic understanding of target objects) [95, 96] or with spatial location on the retina (as more peripheral locations suffer from degraded resolution and crowding effects) [97]. However, if saliency is at least being influenced by objects, this seems to prompt a sort of chicken-and-egg problem in which one asks whether an object is being noticed because it is salient, or whether a location is salient because an object is noticed there. In order to deal with this issue, we need to question what the role of saliency might be within general vision, particularly in light of the temporal characteristics of human object detection.

1.2.2.4 Saliency as an Attention Module

As mentioned in Section 1.2.2.1, RSVP is a style of psychophysical experiment that displays a train of images to a subject with very short temporal display windows [73, 71, 72]. Humans are consistently shown to be able to interpret visual stimuli at an above-chance level even for time intervals that are below the necessary duration to initiate eye movements. However, saliency algorithms have a tendency to perform poorly when predicting the most relevant parts of the target images for processing [74, 97, 75]. These results suggest that human performance is not based on saliency-based gating of visual data, and therefore that saliency does not actually serve as a general form of attentional gating that is universally applied to incoming data. Perhaps the most useful perspective for organizing thinking along these lines is that of Cognitive Programs (CPs) espoused by Tsotsos and Kruijne [98] that attention is not one monolithic process, but rather is an abundant collection of processes that can be dynamically combined according to the requirements of the task. This is a modern update of Ullman's seminal work on Visual Routines (VRs) [99], particularly with respect to the role of attention and perception. In VRs percepts were assumed to have been successfully and correctly calculated, and VRs focused on the subsequent combinatorial strategies for further processing. CPs extends this work to also include fundamental perceptual routines, including attention and saliency.

One of the benefits of the CPs outlook is that it encourages a fundamental analysis of the task itself, which has important ramifications for the role saliency might play. In the RSVP task, there is no requirement for spatial localization, and not enough time for eye movements. Additionally,

Potter [71] found that memory and recognition of a specific image was quite poor compared to conceptual recognition (i.e. Did you see *this* boat? vs. Did you see *a* boat?), suggesting that conceptual recognition can be performed with a rapid feed-forward sweep whereas specific recognition requires further processing, such as feature binding. A great deal of physiological evidence exists for distributed parallel processing of numerous visual features in the visual hierarchy of primate vision [46]. Feature binding refers to the association of multiple visual features into a coherent representation of the stimulus, such as binding object shape with colour and texture (for a comprehensive overview, see [48]), which is likely necessary to consolidate memory of a specific instance of an image. This difference in processing requirements between specific and conceptual recognition is consistent with Koch and Tsuchiya's [100] perspective that subjects can become aware of the gist of a scene without the use of top-down attention, except it includes the stronger claim that this performance necessarily is not spatially gated by an a priori bottom-up attentional map.

However, if visual saliency is not a universally applied pre-attentive processing step, this raises the questions of when and how it contributes to visual processing. Tsotsos [19] and Tsotsos and Kruijne [98] have attempted to formulate this role as one of a number of representations that contributes to the determination of spatial fixations. Such an approach is consistent with saliency not playing a role in tasks that are relatively independent of feature binding, but suggests that it is nevertheless a crucially important attentive mechanism for a wide range of visual processing tasks. This perspective is the basis for Chapter 5, in which saliency is integrated as a component of a larger gaze control system.

1.2.2.5 Saliency and Visual Search Tasks

One vitally important aspect that needs to be discussed when formulating the role of saliency in visual cognition, as well as when determining how to test the efficacy of models, is to clearly articulate the task under which the role of saliency is being examined. As was mentioned in Section 1.2.2.2, one of the primary research avenues along which saliency algorithms are developed and tested is fixation prediction, specifically for predicting human fixation locations under free-viewing conditions (experimental conditions in which no task directions are given to the subject observers). However, there are several issues with focusing saliency model output so heavily on fixation prediction. Much of the earliest formulation for the concept of saliency was rooted in the

psychology of visual search (*e.g.* see [101], as well as the psychophysical evaluation prominent in early models of saliency [25, 64]), and while visual search tasks vary in the degree of task direction given to subjects and may approach free-viewing as instruction is minimized, they nevertheless represent different behavioural contexts, and an overemphasis on performance in one domain risks neglecting performance in the other (see Sections 4.1 and 4.2 for further discussion and experiments on this topic). Additionally, free-viewing does not inherently mean that a subject is reacting solely to the visual stimulus presented; contextual and world-knowledge may easily factor in (particularly for dynamic stimuli; see Section 1.2.3), as well as information gathering strategies and heuristics (see Section 1.2.2.6). This ambiguity between stimulus-driven attentional pull and other potential decision factors further motivates the view advocated for in Section 1.2.2.4 that saliency is best viewed as one component of a larger attentive network, and the prior research that examines the interaction between saliency and visual search task considerations is briefly reviewed here.

The fact that task influences human fixation patterns was established with the earliest eye tracking studies [10], and evidence that it influences endogenous allocations of covert selection was found even earlier [35]. However, the degree to which task considerations dictate human attentional selection is heavily debated, and appears to be a function of many factors including, for example, temporal factors [102]. Saliency appears to aid in finding a target even when a target is defined along a different feature dimension [101], highly salient distractors may [103] or may not [104] adversely affect a subjects' ability to find a specified target.

This relationship grows even more complex in natural images. Zelinsky *et al.* [105] claim that for target search over natural scenes, any bottom-up contributions worsened performance, leading them to claim that top-down considerations dominate visual search. However, in this particular experiment top-down was defined as a template search for the target, while the target was not necessarily salient along any feature channels incoporated into the bottom-up saliency model compared against. Thus, it seems completely natural that the top-down version of their model would perform the best, as it is not solely top-down but rather represents a bottom-up search in which feature conspicuity is maximally tuned by top-down modultion to the specific target being searched for. It is worth noting that Navalpakkam and Itti [49] pointed out that for an environment in which much of the background or other potentially distracting scene elements share similar characteristics with the target, this maximal tuning may not actually be an optimal

choice.

The template search approach was extended in the Target Acquisition Model (TAM) [106] and again shown to be an effective predictor of eye movements during visual search over a wider range of stimuli. Hamker [107] provides another visual search model that operates over natural images. Hamker's model differs from TAM in its emphasis on linking components to the neural circuitry for eye control, and the use of a neural response representation for the target object rather than retention of the full target template. While Hamker's model does not fully retain the target template, both TAM and Hamker's model are structured in a manner that requires an explicit target. However, there is also evidence that such explicit target knowledge is not required for some top-down tuning to provide input to visual search. When less information is known about a target (*i.e.* the target category is provided rather than a visual preview of the specific target [108]) the length of time for search and the number of eye movements necessary to find the target increases, but there nevertheless still appears to be a benefit over what would be expected in the absence of guidance. Likewise, scene understanding can contribute to search guidance, but inconsistent locations will still frequently be searched once expected locations have been exhausted without discovery of the target [109]. This guidance of eye movements by contextual scene knowledge is actually consistent with the idea of visibility models (reviewed in more detail in Section 1.2.2.6), which posit that much of fixation selection is based not on the visual appearance of the scene, but rather is driven by an internal strategy for information gathering.

Thus, while task knowledge appears to be a powerful attribute in guiding eye movements during visual search, it appears to operate over a sliding scale of effect, and one should not completely eliminate the bottom-up, stimulus-driven pull of saliency. Depending on the structure of the visual search task, observers may be able to complete it entirely based on the stimulus attributes without any explicit target knowledge (*e.g.* the singleton searches used in [110] and [111]), they might use incomplete or contextual knowledge to improve search performance [109, 108], or they might appear to completely override considerations of salience and rely on endogeneous directions [112, 113]. This variability in the contribution of saliency to the completion of a visual search task may in part explain the shift away from visual search evaluation of saliency models and a much heavier focus on fixation prediction in free-viewing, but this is unfortunate as a complete explanation of visual search behaviour cannot be done without considering (and effectively modelling) the contribution

of salience. Sections 4.1 and 4.2 therefore provide an examination of the ability of a number of saliency algorithms to account for human visual search performance, while Chapter 5 presents a comprehensive eye movement control model incorporating saliency-driven conspicuity. While the model presented in Chapter 5 addresses a similar problem to the one approached by Hamker's model and TAM, the focus of Chapter 5 is on free-viewing with the aim of getting this aspect of behaviour well modeled first. The architecture may then be extended in future work to incorporate task influences. This will allow for a more complete model of saccadic behaviour whether or not a target template is available.

1.2.2.6 Visibility Models

One avenue of research that should be noted is an alternative view of saccadic targeting. Saliency algorithms make the fundamental assumption that saccades are designed to bring targets of interest (which stand out from their surroundings in a purely stimulus-driven manner) into the foveal region for better analysis. While this may frequently be a valid assumption, there is evidence that it is not always the case, and endogenous considerations must also be taken into account. Foulsham and Kingstone show that when the subject's view of an image is restricted by a gaze-contingent box (a sub-window of the image that translates along with their eye movements), a significant number of saccades are made outside of the viewable area [114]. Saccades outside of the viewable region must clearly be targeted according to some consideration other than stimulus saliency.

Unlike saliency algorithms, visibility models seek to explain saccadic patterns based on how a selected target location will improve task performance and information gathering [40]. Najemnik and Geisler [115] propose a model of ideal observer movements based on an ideal foraging strategy in a cluttered environment. They provide evidence that human eye movements are better matched to this strategy-based approach seeking optimal information gathering given retinal drop-off in acuity outside the fovea than they are to WTA selection of local targets. Of course, this model is not applicable to complex tasks in which targets must be frequently returned to such as when operating in a dynamically changing environment (*e.g.* while driving [116]) or due to the need for highly precise visuomotor feedback (like when threading a needle [117]). Interestingly, Steinman et al. [118] demonstrated somewhat counter-intuitively that saccade target variability decreased with worsening acuity, suggesting that subjects looked only as close to a target as necessary to complete

a task. This has important ramifications for saliency map verification that uses human fixations as ground truth, suggesting that not all fixations are as strictly targeted as others.

However, one major drawback of the visibility model paradigm is that it fails to explain cases of automatic attention capture and response time disruption due to the presence of irrelevant but highly salient targets. The degree to which these occur appears to be highly task-dependent, but for some visual search tasks it is a significant factor in performance [119, 51]. Thus, it would appear that the most comprehensive model of human gaze control will necessarily incorporate these higher-level, strategic considerations, but cannot be made completely independent of stimulus-driven effects. Fecteau and Munoz [59] propose the term priority map for this sort of hybrid selection scheme for saccade targeting. If human gaze prediction is to eventually move beyond free-viewing conditions, and possibly even if we hope to continue improving performance within free-viewing conditions, it is likely that the field must make a concerted shift to priority map models.

1.2.3 Saliency in Dynamic Stimuli

Applying saliency mechanisms to dynamic stimuli (such as video or a live camera feed) rather than static images in many ways changes the underlying problem. Although many of the same principles of attraction that apply in static images will also be present, dynamic stimuli additionally introduces the challenge of prediction (attending to a location where a viewer expects something salient to shortly appear), tracking salient movement, and updating the representation of saliency in time with changes to the stimulus. A number of efforts have been made to extend saliency algorithm analysis from static images to video. Although some naïve approaches simply apply a standard saliency algorithm to each individual frame, most algorithms designed for dynamic stimuli extend the feature representation of the algorithm to include spatio-temporal motion filters (see, for example, [120, 121, 122]). This extension is vitally important given the well-documented influence that stimulus onset and motion have on attentional capture [123], but nevertheless appears to be insufficient to bring saliency performance on dynamic stimuli to the level achieved on static stimuli.

Videos will frequently show a much higher inter-observer consistency than in static images [124], and yet usually yield much poorer saliency algorithm performance when compared against static datasets [125, 126]. Loschsky et al. [127] argue that this is because video clips automatically elicit endogenous attentional cues based on narrative structure, something which is beyond the scope

and capacity of most saliency algorithms to model. In order to overcome this lack of semantic understanding, either saliency algorithm complexity will have to greatly increase (to the point that it little resembles the original intent of saliency research), or the saliency map will likely need to be integrated as a module within a larger system, such as in the form of a priority map proposed by Fecteau and Munoz [59] (in which case the top-down modulation will need to be fully implemented) or in the multiple representation system outlined by Tsotsos and Kruijne [98].

1.2.4 Saliency Applications in Computer Vision

Prior to this point, the discussion has largely focused upon saliency's role in human vision. However, saliency methods are actually much more generally applicable, and a number of works have already explored different ways of incorporating saliency-like calculations. Briefly reviewed here are applications in image quality assessment (IQA), anisotropic image compression, robot nagivation, and the soft-weighting of features. Nevertheless, as mentioned in Section 1.2.3, the primary focus of this work is on the role of saliency in modelling human visual capabilities, and this section should therefore not be viewed as a comprehensive or exhaustive review of the area.

1.2.4.1 Image Quality Assessment

Digital images are frequently subjected to post-processing procedures including compression, format conversion, and re-sizing. Each stage of processing might introduce image degradation or artifacts, but quantifying the change in quality of an image following processing is remarkably challenging. Part of the reason for this challenge is that objective changes in image properties do not always correlate with the subjective quality experienced by human observers who are judging an image [128]. This is in part due to the general robustness of human vision to small degrees of visual changes, but is further exacerbated by the fact that even changes that are objectively the same may not be equally likely to be noticed depending on where in the image they occur. Distortion to heavily attended regions are more likely to be noticed than those that occur in less focused areas of the image, prompting efforts to modulate the prevailing objective measures of changes in image quality by the integration of a saliency map [129, 130, 131].

1.2.4.2 Anisotropic Image Compression

In many ways a complement to quality assessment, anisotropic compression seeks to leverage the fact that human observers are less likely to notice compression artifacts at unattended or inconspicuous locations than those that occur in salient regions. Thus, given a saliency algorithm with adequate predictive powers, large savings in storage and transmission bandwidth can be achieved with relatively little observed loss in image or video quality by applying differential levels of compression between regions of high and low salience [132, 133, 27]. This is particularly valuable for video compression, as online streaming of entertainment now accounts for approximately seventy percent of peak internet traffic in North America [134]. However, as mentioned in Section 1.2.3, saliency prediction in video remains a significantly more challenging problem than for static images. One route of possible promise is to incorporate top-down object guidance with traditional bottom-up saliency methods to improve saliency prediction and thus apparent quality after anisotropic compression. Harding and Roberston [135] demonstrate the efficacy of such an approach to static image compression, but to be effective for video compression it would require a method for determining the probable object of focus in each frame.

One additional aspect of anisotropic compression that should be noted is the need to balance compression levels with the level of inter-observer consistency. As mentioned in Section 1.2.3, video sequences have a tendency to have greater consistency in fixation points across observers, yet even then the degree of consistency may vary across frames, or in conjunction with the understanding of the audience [127]. The effective development of a system for anisotropic compression will therefore most likely require both high accuracy in the prediction of salient regions as well as an assessment of inter-observer variability in conjunction with that salient prediction. This might, for example, allow compression rates to reduce over frames for which confidence in observer adherence to the distribution of salient regions is low, while highly confident frames can nevertheless allow for strong compression rates to achieve storage and transmission targets.

1.2.4.3 Robot Navigation and Camera Control

A common processing step in robot navigation based on visual processing is the need to match features across image frames (both temporally and in stereo depth perception). However, as the

number of features increases, there is a significant increase in both the computational complexity of the feature matching problem as well as the difficulty of accurately pairing features. Computational complexity is of particular concern in mobile robotics, where one is frequently dealing with limited processing power due to embedded hardware but there is still the need to achieve real-time image through-put to deal with dynamic environments. Therefore, one possible approach to mitigating this issue is to pre-screen the features in some way and only select a subset of particularly useful or high-quality features that will go through the feature matching stage of processing. A thresholded saliency map provides one relatively straightforward approach, as most traditional saliency algorithms are designed to require only a small number of processing stages, and conspicuous image regions may be reasonably expected to yield higher quality navigational features [28]. The efficacy of this approach may potentially be further increased by developing an application-specific notion of visual saliency [29].

An alternative application of saliency to robotics is in the active control of the robot's sensors. Early work in this vein was performed by Clark and Ferrier [22], who developed a general-purpose controller for a binocular vision system. More recent work on robot navigation is presented by Rasouli and Tsotsos [30], who use visual saliency to modify the probability space for autonomous visual search. They extend the active search work of Shubina and Tsotsos [136] and Saidi *et al.* [137] in which a robot maintains a probability cloud representation of possible object locations in an environment to more efficiently guide its search. However, rather than discarding information beyond the robot's effective depth of field for object detection, Rasouli and Tsotsos obtain a saliency representation of those image regions to modulate the a priori values of the probability cloud. Such an approach is not inherently limited to robot visual search, but could potentially be utilized in other problem domains as well in which some visual information is present but not enough or of too poor quality to make a firm decision.

1.2.4.4 Soft-Weighting of Features

In some problem domains it is not always desirable to prune features, but it nevertheless may be advantageous to differentially weight the impact of features depending on their importance to the current task. As a representation of the most important or conspicuous information, saliency represents a possible candidate for dynamically determining such a weighting. The effectiveness of

this approach was demonstrated by Feichtenhofer *et al.* [138], who used a spatiotemporal model of saliency to weight the importance of features that were pooled into a bag-of-words (BoW) representation for action recognition. This soft-weighting scheme greatly boosted classification performance on several action recognition databases.

1.2.4.5 Further Applications in Computer Vision

There are two main patterns to the application of saliency to classical computer vision problems presented above: saliency can be used to reduce the computational processing load and to improve the dynamic utilization of available information. This is based on a rather broad definition of saliency as a computationally efficient heuristic for identifying important information. However, not all problems will deem the same information to be of equal importance, and it thus becomes challenging to translate saliency research from one problem domain to another. Nevertheless, progress on saliency prediction in one domain may be useful in a number of other domains. In order to take the fullest advantage of saliency advances there are two important considerations: access to saliency model implementations with as little impediment to experimentation as possible, and the development of performance metrics and datasets that ensure that model evaluation measures the desired algorithm characteristics as accurately as possible. Chapter 2 provides a critical analysis of commonly used fixation-based metrics and proposes a novel evaluation metric for exploring the issue of spatial bias. Chapter 3 presents a software library providing a common access format for a wide array of available saliency models, allowing for rapid application and experimentation.

1.2.5 Review of Saliency Algorithms

Although not an exhaustive list, this section provides a brief summary of a wide cross-section of saliency algorithms that have been developed. These algorithms are broken into a number of loose classes, although the properties of algorithm classes are certainly not mutually exclusive and the predominant features leading to selecting one class over another for a particular algorithm may differ in other analyses. Note that I have primarily focused here on *appearance* based saliency models, which largely act upon features assigned to individual pixels. There is additionally a number of models that focus on saliency based on object detection (*e.g.* [139, 140]), largely based on the proposal by Einhäuser *et al.* [91] that objects predict fixations better than pixel saliency. This claim

has proven controversial, however (see [94] for a rebuttal). In the interest of providing a thorough review of stimulus based saliency, the focus here is on reviewing algorithms that produce pixel-level assignment of saliency values (but refer to Section 1.2.2.3 for an overview of the objectness perspective). A comparative ranking of performance attributes for the models reviewed here is provided in Section 1.2.5.6.

1.2.5.1 Prescribed Feature Processing Models

Prescribed feature processing models represent the earliest class of saliency model. These models largely consist of hand-picked, parametric filter kernels that are selected to resemble the early features in human visual processing. The response output from the filter processing is then combined in a manner that again tends to be loosely based on biological feature channels. While biological fidelity is rarely exhaustively explored or applied as an exact constraint, it is still a distinct motivating factor in model development.

1.2.5.1.1 The Itti-Koch-Niebur (IKN) Saliency Model is one of the earliest saliency models. First described in [25], the model takes colour images as input. An input image is projected across a Gaussian pyramid to create nine spatial scales. Using across-scale differences to perform centre-surround operations, a total of 42 feature maps are created. Six of those maps correspond to feature maps extracted from different intensity scales, 12 are for red-green and blue-yellow colour double-opponency feature maps, and 24 are orientation feature maps created using Gabor filters applied to the intensity pyramid. In the absense of top-down direction, each feature map is normalized by performing the following operation:

1. Normalize the values in the map to a fixed range $[0, ..., M]$
2. Find the map's global maximum M and compute the average $\bar{m}$ of all other local maxima.
3. Globally multiply the map by $(M - \bar{m})^2$

The normalization operation serves to smooth out the feature maps, reduce the impact of maps that respond to numerous elements in the scene, and accentuate distinct feature peaks. After normalization, feature maps are combined into three conspicuity maps along the intensity, colour,

and orientation channels. These conspicuity maps are then themselves normalized and averaged to form the overall saliency map.

In its original formulation, the saliency map was then fed into a two dimensional winner-take-all (WTA) neural network. When one of the WTA neurons reaches threshold, it triggers three simultaneous mechanisms:

1. Focus of attention shifts to the location denoted by the triggered neuron
2. Global inhibition resets the activity of all WTA neurons
3. Local inhibition is transiently activated in the saliency map centred on the locus of attention (an IOR mechanism)

However, modern applications rarely explicitly model these fixation saccades. Instead, most standard scoring metrics (see Section 2.2 for a discussion of scoring metrics) simply take the saliency map as input rather than an explicit set of predicted fixations. Likewise, smoothing or blurring of a saliency map has become a standard post-processing procedure, so modern implementations of the IKN model will usually include by default a final smoothing step [141].

1.2.5.1.2 Graph-Based Visual Saliency (GBVS) is in many ways a successor to the IKN model, and seeks to take advantage of established computational methods in graph theory to provide a robust combinatorial method for integrating local feature responses [78]. Although the method itself is fundamentally agnostic to the specific feature set used to analyze the image, in its standard formulation it uses very similar features to the IKN model. The original paper and, at the time of this writing, default code package [141] includes the following filters at three down-sampled spatial scales:

- Pixel intensity along each channel of the Derrington-Krauskopf-Lennie (DKL) colour-space [142]
- Pixel intensity of the total luminance channel
- Gabor-filtered orientation maps at four orientations: $\{0°, 45°, 90°, 135°\}$

For each filter an activation map is computed. This is done by treating each pixel as a node in a graph corresponding with the same topographic graph location as it has in the image. In order to

speed up calculations, the filter response images are heavily downsampled, with a default maximum graph dimension set to 32. The nodes of the graph are fully connected, with edge weights between each node pair assigned according to the product of the dissimilarity between the node values (filter response) and an inverse distance measure between nodes (closer nodes will tend to have higher edge weights). The outbound edge strength from each node is normalized to 1, thereby defining a Markov chain over the graph. The activation map is set to the equilibrium distribution over the Markov chain.

In the original formulation of GBVS an additional normalization step is computed by re-weighting the graph according to the node activation values rather than node dissimilarity and then once again finding the equilibrium distribution over a Markov chain on this graph, but the default settings of the released implementation instead normalize the activation map by simply raising all values to a power of 1.5 (since all activation values are between 0 and 1, raising by a power greater than 1 will preferentially preserve larger activation values).

Once all activation maps have been computed and normalized, activation maps are averaged across each feature channel to create a set of feature maps. These feature maps are then linearly combined into a total saliency map, which is up-sampled back to the original image size. As with IKN, a post-processing smoothing step has been added to modern implementations of the model.

1.2.5.1.3 Context-Aware Saliency (CAS) is a method that attempts to merge a number of different cues into a single saliency map [143]. The model considers the following major contributors when calculating saliency: low-level factors such as contrast and colour, global factors of feature frequency, visual organization factors (namely the prior that salient pixels should be clustered together), and prescribed high-level features (in the original implementation this is limited to faces).

Features in CAS consist of vectorized image patches in CIELAB space. CIELAB is a colour space that attempts to match colours according to perceptual uniformity, such that distance within different parts of the colour space correspond to roughly the same perceptual change [144]. This allows CAS to compare patches by taking the Euclidean distance between them and expect the distance measure to have close to equivalent meaning regardless of the specific pixel values within the patches. High salience scores are assigned to pixels that are the centre of patches that have

low similarity to other image patches. This calculation is carried out at multiple spatial scales and averaged. A threshold is then applied (0.8 for a map over the range 0 to 1 in the default implementation) to calculate candidate "foci" pixels, and the saliency of non-foci pixels is degraded based on their distance from the nearest foci pixel. Face detection is incorporated as a post-processing step in a binary fashion, and pixel saliency is taken as the maximum between the feature saliency and the output of the face detection.

1.2.5.1.4 Covariance Saliency (CVS) takes a local patch-based approach to saliency calculation [145]. The image is split into non-overlapping rectangular image regions, and each patch is compared against its respective neighbouring image regions using the region covariances descriptor developed by [146]. Covariance descriptors were chosen both for their compactness of representation and their efficiency of computation using a second-order integral image representation. Additionally, because covariance matrices encode the interactions of their constituent feature elements, they provide a natural mechanism for combining visual responses across feature channels without relying on linear averaging or ad-hoc weighting.

CVS calculates a covariance feature matrix using a seven-dimensional feature vector over each pixel consisting of: CIELAB colour values, edge-orientation in the vertical and horizontal directions, and the raw pixel location. The covariance descriptor is additionally augmented by first-order image statistics (feature means) to be sure to capture salient changes in overall patch properties in addition to the textural properties encoded in the covariance matrix. CVS additionally directly encodes a centre bias term by linearly penalizing the saliency of image patches as one moves further from the centre of the image. To account for spatial scale, CVS is calculated over a set of image patch sizes and then recombined by taking the point-wise product of each spatial scale.

1.2.5.1.5 RARE is a series of algorithms that have developed as successive modifications to a processing pipeline. Although primarily based on the idea of feature rarity as a driving force for saliency based on the self-information measure first proposed in the AIM model [2] (see below), the algorithms nevertheless provide a prescribed pipeline based more on empirical tuning of results than an information theoretic foundation.

In the first full incarnation of the RARE algorithm [147], an input image is converted from

RGB colour space to YCbCr colour space. YCbCr is a colour space representing the luminance (Y-channel) along with two colour opponency channels (Cb and Cr) that generally provides a more decorrelated representation of the chromatic values of an image than the traditional RGB representation [144]. The image is then run through the following steps:

1. Apply a Gabor filter bank representing a range of orientations and scales to each colour channel

2. Within each colour and orientation channel, average filter responses over all spatial scales to create filter response maps

3. Each filter response map is quantized into C values, where C is a parameter of the algorithm, according to an iterative quantization algorithm, each one representing a feature map

4. A binary mask representing a feature map is then projected across a pyramid

5. The value of each pixel, normalized across the pyramid scales, represents the probability of a given filter response incorporating both local and global contextual properties

6. The self-information of each filter response map is taken to form a conspicuity map along that feature channel

7. Filter response channels are then combined within each colour channel by a weighted linear combination, where channel weight is determined by a ranking of those channels with greater distance between maximal and mean response

8. Colour channels are now combined according to the same principle as in step 7

An updated version of the RARE algorithm, dubbed RARE2012, was subsequently released that modified the above pipeline [148]. Rather than map images into a standard YCbCr space, RARE2012 instead utilizes a maximally decorrelated space through Principal Component Analysis (PCA) decomposition on the RGB values. In addition to the oriented Gabor filter channels, RARE2012 also performs the same feature analysis on the raw pixels in each colour channel. Rather than a direct average in Step 2 to create filter response maps, a weighted average is done using the same principle as in Step 7, but modified (along with the averaging done in Steps 7 and 8) to include a threshold

to eliminate completely the influence of any feature maps that fall below a certain threshold. The quantization in Step 3 is replaced instead with a simpler histogram-based probability density estimate. These changes lead to improved performance over the original RARE pipeline for all metrics and datasets tested by the authors.

An additional modification to the basic pipeline was introduced with the SuperRare algorithm [149], which uses superpixels to introduce the notion of object-based salience to the RARE pipeline. SuperRare removes all orientation filtering from the algorithm, and instead produces saliency based soley on the colour information in the image (as segmented by superpixels). Preprocessing filtering is done to improve superpixel segmentation, and the image is projected into a large number of colour spaces (RGB, Lab, Luv, HSV, HL5, and YUV).

1.2.5.1.6 VOCUS2 is a recently released updated version of the original IKN model [25] developed by Frintrop et al. [150]. It aims to harness the intuitively simple and biologically inspired structure of IKN with increased performance based on subsequent developments in the field. The overall structure is the same as IKN, but with the following distinctions:

- In original IKN there is a colour channel and an intensity channel. In VOCUS2, this has instead been replaced by the use of a simple colour-opponency space $[I, RG, BY]$ in which $I = \left(\frac{R+G+B}{3}\right)$, $RG = R - G$, and $BY = B - \frac{R+G}{2}$. Each channel in this colour space is treated separately before the final saliency map integration, rather than combining colour information into one feature channel.

- The Gaussian pyramid structure of IKN is replaced with a more sophisticated scale-space processing structure based in part on Lowe's work [151]. Instead of a single pyramid structure, two pyramids are built consisting of *center* pyramid $C = (C_0, ..., C_k)$ and *surround* pyramid $S = (S_0, ..., S_k)$. Surround layers are derived from their corresponding center layers by smoothing with a Gaussian kernel. The size of this kernel is based on the desired center-surround ratio, thereby allowing for more control over the center-surround operations than is possible within a single pyramid.

- Optional removal of the orientation channel. VOCUS2 was initially developed as a prototype object segmenter, and it was found that including the orientation channel tended to pull

saliency toward the edges of the object rather than filling in the object centre. However, the authors concede that for other applications it may be prudent to re-add orientation information.

- Optional addition of a spatial prior.

Given its relative newness and lack of inclusion in the MIT Saliency Benchmark [79] project, VOCUS2 has little performance data available for it. Nevertheless, the evaluations that are available show promising results not only in fixation prediction, but also in the domain of salient object detection [150].

1.2.5.1.7 Boolean Map Saliency (BMS) is a method that combines several heuristics and mathematical abstractions to create a computationally efficient yet surprisingly effective method for saliency prediction [152]. It focuses on figure-ground segmentation, and thus tends to produce maps that concentrate saliency more toward the centre of objects than the boundaries without the need for additional smoothing, and has thus also found success in salient object segmentation.

The method works in three stages of map generation before merging into a final saliency map: Boolean maps, Activation maps, and Attention maps. Boolean maps are created by a randomized distribution of thresholds over a set of feature channels; in the default operation the channels are simply taken as the colour channels in CIELAB colour space. Activation maps are calculated by removing regions that lack the surroundedness property (*i.e.* regions that touch the image border); this is based on the assumption that salient objects will be fully within the image frame. Attention maps are then calculated by applying the L2-norm to each activation map, thereby placing emphasis on small regions of concentrated activation preferentially over maps with highly distributed activations. Attention maps are then averaged to produce the final saliency map.

Each step in the algorithm can be rapidly calculated on a computer even with only limited hardware (such as without a GPU), thereby allowing BMS to rival many of the spectral methods detailed below for speed. It likewise tends to outperform most spectral methods, albeit it makes very strong assumptions about photographic composition that may cause issues of generality should a wider range of input images be used. This is of particular issue given that many applications for which the method's low hardware requirements and fast processing speed would be most attractive

are for embedded applications such as mobile robot platforms, but for such systems the assumption of a well-composed image in which objects of interest do not intersect the image edge may be least likely to hold.

1.2.5.2 Information Theoretic Approaches

Information theoretic approaches focus on the problem of saliency prediction with a primary focus on the problem in terms of information processing. Although these methods tend to emphasize a principled mathematical formulation of the problem of saliency prediction, they frequently will still motivate this formulation on and compare its resultant behaviour with human physiological and psychophysical results. This corroboration with human data is based on the proposition that a well-formed description of the problem from first principles will result in a model with many of the same emergent properties as those exhibited by human vision.

1.2.5.2.1 Attention by Information Maximization (AIM) is one of the first saliency algorithms constructed from mathematical first principles by trying to solve the problem of conspicuity labeling rather than explicitly modelling a biological process [2]. Nevertheless, the authors retain an interest in biological fidelity and in subsequent publications have explored the relationship between the AIM algorithm and results in both neurophysiology and psychophysics [153, 154, 155, 156, 157]. In addition to releasing a novel approach to saliency modelling, the authors of AIM additionally released an early dataset of images with corresponding human eye tracking fixations commonly referred to as the Toronto dataset [1]. This dataset has seen heavy usage in saliency algorithm testing (*e.g.* see [1]).

The basic intuition behind the model is that the most salient part of an image is that which was least easily predicted. To achieve this, AIM starts by seeking a sparse code for natural image statistics. In the original formulation this is created by performing Independent Component Analysis (ICA) over a large collection of randomly sampled image patches to generate a set of feature bases covering the image space. After training the ICA basis, AIM operates as follows:

1. Project the image onto the ICA basis by convolving each ICA patch with the image (the ICA patches are treated as a filter bank of convolution kernels)

[1] Available for download from: `http://jtl.lassonde.yorku.ca/software/datasets/#fixation_data`

2. For each ICA feature, calculate a probability density estimate over the range of possible filter responses by creating a histogram of the specific pixel responses generated for that feature. In the original formulation of AIM a separate histogram was generated for each local neighbourhood of the image, but for computational simplicity most implementations of AIM simply calculate the probability density estimate over the entire image

3. For each pixel in the image, the joint probability of its projection into an ICA space with N basis functions is given by $P(r_1, r_2, ..., r_N)$ where $P(r_i)$ is the probability of the response of the pixel to filter i according the the probability density estimate generated in the previous step. By assuming independence of the filters, the joint probability can be simplified to the product of individual probabilities: $P(r_1, r_2, ..., r_N) = \prod_{i=1}^{N} P(r_i)$

4. Saliency is calculated by taking the self-information of the pixel response probability, which is equal to $-\log(p)$. The saliency value for each pixel thus becomes:

$$\textbf{SALIENCY} = -\log(\prod_{i=1}^{N} P(r_i)) = -\sum_{i=1}^{N} \log(P(r_i)). \tag{1.1}$$

It is useful to note that the actual choice of features in the model is relatively unconstrained. So long as the features cover a sufficiently broad space of image properties and are approximately independent they should provide a functional basis for AIM. Testing of a number of commonly used image filters has shown that some parametric filters, such as log-Gabor filters, can achieve similar levels of performance [153] to ICA-generated features. This is perhaps unsurprising, given that ICA training often yields results resembling a bank of oriented bandpass filters [158].

1.2.5.2.2 Saliency Using Natural Statistics (SUN) uses a Bayesian framework to characterize saliency [159]. Although they include terms through which top-down knowledge regarding object class and/or location may be included, in the original formulation (and standard application) of the algorithm those terms are left undeveloped and only the bottom-up term is utilized. The authors therefore converge on a very similar definition of saliency to that of AIM (see previous) in

which, also assuming feature independence, saliency is defined by the equation:

$$\textbf{SALIENCY} = -\sum_{i=1}^{N} \log(P(F_i = f_i)), \tag{1.2}$$

where $P(F_i = f_i)$ is the probability for a given pixel that the ith feature takes on the observed value of f. In order to maximize feature independence, the features used in SUN were also derived by training an ICA basis set. However, the critical difference between SUN and AIM is that in SUN the expected probability distribution for a given feature is learned rather than estimated based on the other responses within an image. This is based on the assumption that saliency is driven by prior experience, and justified by the fact that it can easily capture asymmetries in visual search performance (such as it being easier to find a magenta dot amongst blue distractors than a blue dot amongst magenta distractors). This also provides a computational advantage over AIM by reducing the probability density estimation step to a single probability look-up rather than a global histogram calculation. However, by pre-calculating the expected probability distribution over each feature, SUN becomes unresponsive to important contextual cues in a given image, introducing new search asymmetries that are non-existent with human observers (for example, if SUN has learned that red elements tend to be more salient than green elements, it will consistently miss a green singleton target amongst red distractors, despite humans having no difficulty with this task).

1.2.5.2.3 Dynamic Visual Attention (DVA) is a model that, similar to AIM and SUN, utilizes a sparse coding representation to calculate saliency [160]. RGB patches are sampled from a training set of natural scenes from which an ICA method is used to calculate a set of basis features. Similar to AIM, the relative activity of these features is calculated over an image, producing a probability function of feature activities. However, rather than subsequently calculating the self information for feature responses at each image location, DVA instead calculates the contribution of feature response to the entropy of the probability function and labels features that increase the system entropy as salient. Crucially, the neighbourhood of the probability function calculation can be extended either spatially or temporally, allowing DVA to relatively easily applied to static or dynamic stimuli.

1.2.5.2.4 Saliency Detection by Self-Resemblance (SSR) uses local steering kernels to characterize the image data, which are then fed into a nonparametric kernel density estimate to derive the likelihood of a local region being marked as salient [161]. Similar to the other information theoretic approaches presented previously, pixel saliency in SSR is intuitively based on how well it stands out from its surroundings. However, SSR differs from previously presented information theoretic approaches by defining saliency as the probability that a pixel should be labeled as salient or not given a particular feature set according to the equation

$$\textbf{SALIENCY} = P(\text{pixel is salient}|\mathbf{F}),$$

where $\mathbf{F}$ is the matrix of feature vectors centered on the given pixel. This feature matrix is formed by a local set of feature vectors and a larger set of feature vectors which is referred to as the surround. SSR makes two assumptions:

1. Every pixel is equally likely to be salient
2. $P(\mathbf{F})$ is uniform over features

Using these two assumptions and Bayes Theorem, the order of arguments in the saliency equation can be reversed such that

$$\textbf{SALIENCY} = P(\mathbf{F}|\text{pixel is salient}).$$

This conditional probability is then estimated based on how similar the local steering kernel is with its surround, where similarity is measured by finding the matrix cosine similarity.

1.2.5.2.5 Adaptive Whitening Saliency (AWS) uses contextual adaption of signal response to derive a compact representation of visual saliency [162] consistent with the efficient coding hypothesis [163]. AWS applies the whitening transformation to the feature vector representing its pixels. A general whitening transformation solves for the unmixing matrix $\mathbf{W}$ given a feature vector $\mathbf{X}$ in the equation:

$$\mathbf{Y} = \mathbf{W}\mathbf{X}, \tag{1.3}$$

such that the correlation matrix of $\mathbf{Y}$ is the identity matrix. For the feature vector itself, AWS uses a discretized Fourier optics representation of the image, assigning to each pixel a feature vector

defined by the combinations of M_λ spectral wavelength components, M_ρ spatial frequency radii components, and M_α possible spatial frequency angles, leading to a vector of total length M where

$$M = M_\lambda \times M_\rho \times M_\alpha. \tag{1.4}$$

In order to reduce the computational load, AWS does not fully whiten the feature vector, but rather hierarchically whitens the chromatic (M_λ) components first, then the spatial components (M_ρ) independently at each orientation, resulting in an approximately whitened feature vector but at a much lower computational cost. They then define an *optical variability*, OV, term as the squared norm of this approximately whitened vector, and define the saliency of pixel p according to the equation:

$$\textbf{SALIENCY} = \frac{OV_p}{\sum_{\mathbf{P}} OV_p}, \tag{1.5}$$

where $\mathbf{P}$ is the full set of pixels.

1.2.5.2.6 Fast and Efficient Saliency (FES) operates similarly to SSR, and uses a sliding center-surround window and Bayesian inference to calculate the probability that a given pixel is salient [164]. Kernel density estimation is used to sparsely sample the windows, thereby reducing the computational load of the model to allow it to be quickly computed. The pixel colour values in CIELAB space are taken directly as the features considered in the center-surround calculation. FES establishes two scale properties: the radius of the circular sliding window, r, and the number of samples, n, taken from the window. Saliency is taken as the average over these scales. Unlike SSR, FES also learns a positional prior, amounting to the addition of a centre bias component.

1.2.5.2.7 Learning Discriminative Subspaces (LDS) is a model that operates by learning a set of representational subspaces that maximally separate the salient and non-salient pixels over a training set [165]. Candidate subspaces are calculated by sampling pixel blocks from training images, vectorizing these image patches, and then performing PCA over the set of patch vectors. The top d (where d is a parameter set during the learning phase) principal components found in this manner are then used to construct a set of candidate subspaces at different spatial scales. Once these subspaces have been constructed, the training images are projected onto them to find the

ones that provide the best contrast between salient and non-salient pixels, and a weight vector is learned over the retained subspaces.

1.2.5.3 Spectral Methods

Spectral methods are unified in their focus on processing in the Fourier domain to leverage aspects of the phase and power spectrum attributes of an image to assign saliency scores. Due to the computational efficiency of the fast Fourier transform (FFT) and subsequent processing in the Fourier domain, one of the primary advantages of spectral saliency methods is that they are generally very fast to compute. Note that a number of approaches described in other sections may make use of filters based on frequency domain analysis (such as Gabor or log-Gabor filters). Here we concentrate on methods that look at more global features in the spectral domain, rather than locally windowed convolutional features.

1.2.5.3.1 Spectrum Residual (SR) was one of the first spectral methods proposed for calculating visual saliency [166]. It uses a downsampled image projected into the Fourier domain to produce a log spectrum $\mathcal{L}(f)$. A generalized log spectrum, $\mathcal{A}(f)$, is calculated as a convolution of the log spectrum with a uniform averaging kernel. The difference between these two spectra is termed the *spectral residual*, $\mathcal{R}(f)$. The spectral residual is then transformed back into the spatial domain through the inverse Fourier transform (IFT) to produce the final saliency map.

The intuitive idea of the SR method is to capture proto-objects by finding unusual deviations within the log spectrum of an image. However, this concept has been rather soundly criticized by Li et al. [60], who replace the spectrum residual by random white noise and produce an almost identical saliency map, prompting them to conclude that it is the phase spectrum, not the amplitude spectrum, that is the primary driving factor in saliency information.

1.2.5.3.2 Additional spectral approaches. There have been a number of additional additions and refinements to spectral methods of saliency calculation since the SR approach [166], such as Quaternion Based Spectral Saliency (QSS) [167], Scale-Space Analysis in the Frequency domain (SSAF) [60], and Image Signature (ImSig) [168]. However, most spectral methods are typically outperformed by the other methods reviewed here in independent benchmarking efforts

(eg. [79, 169, 148]), largely remaining useful as a class of algorithm for which speed and available processing power is an important factor.

1.2.5.4 Motoric Models

1.2.5.4.1 Behavioural Bias Model (BBM) is a statistical model of saccade bias created by Tatler and Vincent [170]. It uses a dataset of human free-viewing fixations to train a classifier distinguishing fixated image patches (**F**) from control image patches (**C**) to learn the log-likelihood ratio:

$$\log\left(\frac{P(magnitude, direction|\mathbf{F})}{P(magnitude, direction|\mathbf{C})}\right). \tag{1.6}$$

They additionally trained similar ratios with added feature (orientation) information as well as another classifier with added saliency information (calculated from the IKN model), but one of their more interesting results is that the log-likelihood classifier based solely on behavioural biases (and therefore fully independent of visual stimuli) significantly outperformed the IKN model's performance on their dataset (area under the curve (AUC) scores of 0.565 for IKN and 0.648 for BBM [170]). This suggests that gaze control mechanisms independent of the visual stimulus are nevertheless likely to be important to explaining human fixation locations. This topic is explored in more detail in Chapter 5.

1.2.5.4.2 Saccade Sequence Generator (SSG) is a model that functions qualitatively similarly to the BB model except that it encodes a more sophisticated set of motor biases that is more directly coupled to the underlying saliency map. The originally proposed implementation by Le Meur and Liu [11] is further improved by an in-depth analysis of context-dependent and spatially-variant viewing biases in saccade control by Le Meur and Coutrot [171]. The model is governed by a conditional probability field modeled as a Markov process of order T such that:

$$p(x|x_{t-1}, ..., x_{t-T}) \propto p_{BU}(x)p_B(d, \phi)p_M(x, t) \tag{1.7}$$

where x is a potential target location, x_{t-i} is the location of gaze i fixations previously, p_{BU} represents the bottom-up (saliency) component, p_B represents the motor bias component (learned as a distribution dependent on the distance d between x and x_{t-1} and the angle ϕ of the vector

joining the two locations), and p_M represents the memory state of location x at time t, and is used to encapsulate inhibition of return and the probability to refixate targets.

Rather than producing a deterministic sequence, the authors instead advocate for a more stochastic approach in which a number of saccade sequences are stochastically sampled from the process. A new static saliency map may then be produced by combining all the predicted fixations together and convolving with a Gaussian kernel (similar to how "human saliency maps" are produced in [4]).

1.2.5.5 Machine Learning Methods

Machine learning approaches typically cast saliency as a classification problem, seeking for each image to provide either a hard- or soft-classification to each pixel as to whether it is salient or not based on a learned model of what counts as salient. Although these models are frequently amongst the top performing algorithms for fixation prediction, they are heavily dependent on the training data and usually lack clear explanatory power regarding *why* a pixel was salient (or not).

1.2.5.5.1 Learning to Predict Where Humans Look (JuddS) is one of the earliest attempts to apply machine learning to saliency prediction, developed by Judd et al. [4]. It was designed purely as a performance model; JuddS provides no mechanism or predictions for the operation of human vision, but rather is explicitly attempting to maximize performance for human gaze prediction. In addition to the development of a machine learning classifier for salient targets, JuddS also released an extensive dataset of images with human eye tracking data, commonly referred to as the MIT dataset[2], providing what was, at the time, the largest eye tracking database for both algorithm testing and training. This dataset continues to see widespread use for saliency model training and testing.

JuddS uses a mixture of what they term low-, mid-, and high-level features. Images were resized to a resolution of 200×200 pixels and for each pixel a feature vector was constructed consisting of the following features:

- Low-level features
  - Steerable pyramid filters over four orientations at three spatial scales

[2]Available for download from: `http://people.csail.mit.edu/tjudd/WherePeopleLook/index.html`

  - Sub-band pyramid features based on the work of Oliva and Torralba [85] and Rosenholtz [172]
  - Intensity, orientation, and colour channels from the IKN model [25]
  - Colour channels based on raw RGB values and an estimated probability of each channel, and the probabilities each colour computed from a 3D colour histograms filtered with a median filter at several different spatial scales

- Mid-level features
  - A horizon line detector based on mid-level gist features also based on the work of Oliva and Torralba [85]
  - Distance of pixel to the image centre

- High-level features
  - Results of the Viola-Jones face detector [173]
  - Results of the Felzenszwalb person detector [174]

Altogether this produces a feature vector of 33 elements. Training sets of 903 images were randomly selected from the full dataset of 1003 images, leaving 100 images for testing. Human fixation data was convolved with a Gaussian kernel to produce ground-truth saliency-style maps for each training image; ten pixels were randomly sampled from the top 20% most "salient" locations in these human maps as positive examples and 10 pixels were randomly sampled from the bottom 70% of pixels as the negative set. The full set of training features were normalized to have zero mean and unit variance; the normalization parameters were then saved as part of the model and applied to the test data. The training data was then used to train a support vector machine (SVM) with linear kernels. Kernels using radial basis functions and multiple kernel learning were also tested, but not found to provide significant boosts to performance. Therefore, linear kernels were retained based on their faster training and more intuitively understood resultant weights.

1.2.5.5.2 Ensemble of Deep Networks (eDN) is one of the earliest saliency approaches to utilize convolutional neural networks (CNNs) to learn a representational set of salient features [3].

In contrast to later CNN-based efforts, eDN uses a set of independently trained shallow networks (maximum depth of three layers) and a final classification step using an SVM. The choice of network shallowness was motivated by computational complexity in order to allow a larger search through network parameter space to find more effective network architectures. Although eDN achieved competitive performance against non-CNN based approaches, it was quickly eclipsed by subsequent models utilizing deeper architectures.

1.2.5.5.3 Deep Gaze I (DGI) is one of the first models to approach the challenge of human gaze prediction using deep CNNs [175]. One of the limiting factors that had previously hampered such an extension of CNNs was the lack of sufficient data upon which such a network could be trained. Deep CNNs require enormous quantities of training data to achieve their results, but once that data is available they can become very powerful predictive models. Building on the work of Barthelmé et al. [176] in modelling eye movements as a point process and Donahue et al. [177] in retraining networks for related tasks, DGI overcomes the lack of training data by repurposing Krizhevsky et al.'s [178] object recognition network (AlexNet).

Rather than object recognition, DGI is instead interested in the spatial location of the high dimensional features represented by the network, and therefore the final three fully connected upper layers of AlexNet (which perform the final recognition steps) are removed. The architecture of the remaining layers is fixed, and DGI trains a saliency classifier on top of the internal filter responses of this network. The authors used half of the MIT dataset produced by Judd *et al.* [4] as their training set.

Each layer of the network is up-scaled to the size of the largest layer in the network, therefore providing every pixel with a response value from every layer in the network. Treating each filter sheet in each layer as a separate element in a feature vector produces a feature vector up to 3712 elements in length if all network layers are included. Each element of the feature vector is normalized to have zero mean and unit standard deviation over the training set. The feature vector is then linearly combined with a set of learned weights and smoothed with a Gaussian kernel. The weights are learned using l_1-regularization in order to encourage sparsity. In order to try and reduce over-fitting, multiple different subsets of network layers were selected for training and the performance was checked on the test set of remaining images from the dataset. A centre bias is added, and

softmax regression is applied to the final resultant map to yield a probability distribution over the image.

1.2.5.5.4 Multiresolution Convolutional Neural Network (Mr-CNN) is a CNN-based model that is directly trained on fixation data (rather than utilizing transfer learning like DGI or the larger SALICON dataset of proxy fixations used in DF and SALICON) [179]. The network is trained by resizing input images to three different spatial scales (400×400 pixels, 250×250 pixels, and 150×150 pixels), and then extracting identically sized image patches (42×42 pixels) from all three scales (with a common centre) to be fed into the network. By using subsampled patches, the number of training samples is greatly increased over just the number of images in the set, thereby allowing for adequate training data for the CNN. Although Mr-CNN demonstrates better performance than most traditional methods against which it was compared, it was only evaluated using the shuffled-AUC method and has not been widely tested in independent benchmarking efforts, making it difficult to fully evaluate its performance within the larger landscape of available models.

1.2.5.5.5 DeepFix (DF) is a CNN-based approach that, in contrast to DGI that used a network trained for object recognition, trains a task-specific network for saliency detection [180]. To overcome the lack of sufficient fixational training data encountered by DGI, DF leverages the SALICON dataset: a novel method of data gathering presented by Jiang et al. [181] that uses mouse-clicks in a foveated image as a proxy for eye-tracking. Training for DF takes place in two stages; the network is initially trained using the large SALICON dataset, and then refined using datasets of actual human eye fixation locations.

1.2.5.5.6 Saliency in Context (SALICON) model [7] is one of the top performing models in the MIT Saliency Benchmark [79]. It is based on leveraging the large training SALICON training set [181] much like the DF model, and has a very similar (albeit slightly better) record of performance, suggesting that it is largely an incremental improvement and refinement of the CNN approach to saliency detection. The model trains two deep networks, one at a fine spatial scale (full image resolution) and another at a course spatial scale (images that have been heavily scaled down in size), with network weight training based on an objective function that seeks to optimize

common saliency evaluation metrics. The authors found that an objective function based on the Kullback-Leibler (KL) divergence (see Section 2.2.1.4) gave the best overall perfromance.

1.2.5.5.7 Deep Gaze II (DGII) is an updated saliency prediction model developed along the same lines as DGI but with a more modern base network [182]. It is still based on repurposing the feature set of a network trained for object recognition, but rather than using Krizhevsky et al.'s [178] network it instead uses features from the VGG-19 network [183]. Similarly to DGI, features from the base network (in this case, a subset of feature layers chosen by random search) are scaled such that they can be combined into a multidimensional tensor and then processed by a readout network that converts responses to the feature detectors into a prediction of saliency. The default action of the model explicitly adds a centre bias as a prior learned over the MIT1003 [4] dataset. Despite the iterative nature of its development, DGII is significant due to its performance; at the time of this writing it still holds top performance for some metrics on the MIT Saliency Benchmark [79].

1.2.5.5.8 Intensity Contrast Features (ICF) is an interesting model related to the development of DGII. Rather than take as input to the readout network features learned for object detection, it instead takes the results of a Gaussian pyramid at five spatial scales across luminance and two colour channels (computed via principal components analysis on the MIT1003 dataset [4]) as the input [184]. The model is therefore restricted solely to a set of low-level, intensity-driven features on which to base its estimation of saliency, making it a model that is much more equivalent to eDN and some of the earlier models not based on machine learning (such as IKN and AIM) in design principles, but nevertheless utilizing the best modern learning techniques. In terms of predictive performance, ICF tends to outperform eDN and models not based on deep learning, but is not typically competitive with models that utilize deep features. Despite being outperformed by several previous models, ICF represents an interesting tool for exploring the impact of feature complexity and showcasing how many modern saliency models are heavily reliant on highly complex feature sets rather than the lower level feature sets that were originally theorized to be part of the early gating role of saliency [25, 185].

1.2.5.5.9 Saliency Attentive Model (SAM) is a model that utilizes modified Long Short Term Memory (LSTM) modules that iteratively refine the spatial allocation of saliency (given that SAM is designed to operate over static images, SAM lacks the temporal component that is typically present when employing an LSTM) [186]. In addition to the network representation, SAM employs a pair of learned central priors as post-processing. Much like other recent high-performing saliency models based on deep learning such as DGII, SAM utilizes the network architecture and learned features from a state of the art model previously employed in object recognition, and then removes the final fully connected layers and instead trains a new set of output layers specific to the task of saliency prediction. In the case of SAM, there are two versions that were developed, one based on the VGG-16 network [183] (SAM-VGG) and another based on the ResNet-50 network [187] (SAM-ResNet). While the two base architectures perform quite similarly, SAM-ResNet typically tests slightly higher on established benchmarks.

1.2.5.5.10 Multi-Level Network for Saliency Prediction (ML-Net) is a model that performs a relatively straightforward chain of processing to compute a saliency map using the features of a CNN [188]. ML-Net uses the VGG-16 network [183] as the basis for its calculations, but modifies the architecture slightly by decreasing the stride employed in the early network layers and thereby reduce the degree of downsampling inherent in the network. Feature output is then taken from three different levels of abstraction and concatenated into a single tensor that is passed through an encoding network to output the saliency map. As with SAM, a spatial bias prior is learned and applied in post-processing.

1.2.5.5.11 Saliency using Generative Adversarial Networks (SalGAN) is a deep learning model that approaches the task of predicting a saliency map over an input image somewhat differently than most other deep networks overviewed here [189]. Rather than apply a loss function using the reported error with respect to a ground truth map derived from the convolution of fixation points with a Gaussian blurring kernel, SalGAN instead makes using of the gnerative adversarial network (GAN) methodology pioneered by Goodfellow *et al.* [190]. In other respects, however, the method operates quite similarly, utilizing the feature layers of VGG-16 [183]. The final two retained feature layers are modified during the training phase of SalGAN, while the earlier layers

are fixed to save on computational resources and training data requirements. As with the DVAP model (detailed below), upsampling is performed using a series of deconvolutions, but unlike with DVAP no skip connections are employed and SalGAN instead relies on a single continuous chain of operations.

Although benchmark performance of SalGAN is comparable to many other deep learning approaches, the generative step is not particularly well justified from a theoretical standpoint, and the addition of this step is likely to further obscure the explanatory traceback (being able to answer why is one visual element is salient while others are not) possible for a given image and its predicted saliency map.

1.2.5.5.12 Deep Visual Attention Prediction (DVAP) is a deep learning model that makes use of skip connections and trained deconvolution kernels to integrate information over different levels of feature abstraction within the network [191]. As with many of the other deep learning based models overviewed here, DVAP utilizes the convolutional layers from an established object detection network (VGG-16 [183]). However, DVAP differs from both SAM and DGII in the nature of the subsequent processing of the object detection features. DVAP employs a decoder network that upsamples the feature maps from earlier in the network using a set of trained deconvolution filters. Despite this more complex handling of feature integration as compared to DGII, on most established benchmark metrics DVAP appears to perform on par with or slightly below DGII.

1.2.5.6 Concluding Remarks on Models

Cross-model comparison is fraught with a number of challenges. It is often difficult to properly probe a model's behaviour and characteristics without access to a functional implementation, but ensuring that a given implementation is set up and running as intended is not always straightforward (see further discussion on this matter in Chapter 3). Nevertheless, Table 1.1 provides a summative assessment of the reviewed algorithms across several avenues of performance (note that performance assessment itself is a matter of some contention, as discussed in Section 2.2, and the values assigned here are therefore based on the personal assessment of the author):

- **Performance:** Assessed from Low to High, this attempts to assess the general level of

performance of each algorithm at predicting human fixation locations. Due to wide range in evaluation metrics and datasets this is necessarily an approximate rating, and primarily refers to performance based on fixation prediction.

- **Robustness:** Supplementary to the performance ranking, this provides an assessment from Low to High on how dependent an algorithm's performance is to dataset- or metric-specific tuning of parameters or training routines.

- **Explanatory Traceback:** An assessment from Low to High of the ability to trace back the resultant saliency map to find an explanation for why a particular pixel or region was more or less salient. Note that this is not a native component of virtually any saliency algorithm, but some algorithms are more or less amenable to this extension based on the way in which data is condenced or abstracted.

- **Reliance on Training Data:** The degree to which an algorithm requires a large corpus of data to function, assessed from Low to High.

There are a few broad trends that appear to hold true across the models presented here. When it comes to application-specific performance for gaze prediction, the current leaders by a relatively high margin are the CNN-based approaches. Nevertheless, given the newness of these approaches and the lack of broad testing in other application areas, they do not seem to be as robustly effective as the more low-level approaches, particularly those methods based on information theoretic principles. Whereas the prescribed feature models will frequently contain high levels of specific optimization built into them, the information theoretic models can often be optimized through post-processing techniques (such as smoothing or applying spatial bias) to rival the performance of these models but, by being largely agnostic to such optimizations in their base formulation, can be more easily applied to a wider range of problems and applications that stretch beyond gaze prediction.

An additional avenue of assessment and comparison is model execution speed. Note that this can be highly dependent on both available hardware and implementation optimization (for example, a number of algorithms employing filter convolutions such as AIM or SUN could likely be significantly sped up by using GPU-based convolutions). It is often difficult to perform comparisons of execution speed from the literature as each particular paper performs tests over different input stimuli over

Algorithm	Performance	Robustness	Explanatory Traceback	Reliance on Training Data
AIM	Medium	High	Medium to High*	Low to Medium*
AWS	Medium	High	Medium-High	Low
BBM	Low-Medium**	-	N/A**	Medium
BMS	Medium-High	Medium-High	Medium	Low
CAS	Medium-High	Medium-High	Medium	Low
CVS	Low-Medium	Medium	Medium	Low
DF	High	Medium	Low	High
DGI	Medium-High	Medium	Low	Medium-High
DGII	High	Medium	Low	Medium-High
DVA	Medium	Medium-High	Medium	Medium
DVAP	High	Medium	Low	Medium-High
eDN	Medium	Medium	Low	Medium
FES	Medium	-	Medium	Medium
GBVS	Medium	Low-Medium	Medium	Low
ICF	Medium	Medium-High	Low-Medium	Medium
IKN	Low-Medium	Medium	High	Low
JuddS	Medium	Medium	Low-Medium	Medium-High
LDS	Medium-High	-	Medium	Medium
ML-Net	High	-	Low	High
Mr-CNN	Medium-High	-	Low	Medium-High
RARE2012	Medium-High	Medium-High	Medium	Low
SalGAN	Medium-High	-	Low	High
SALICON	High	Medium	Low	High
SAM	High	-	Low	High
SR	Low	Medium	Low	Low
SSR	Medium	-	Medium	Low
SUN	Low-Medium	Low	Medium	Medium
VOCUS2	Medium	-	Medium-High	Low

Table 1.1: Table summarizing a number of saliency algorithm characteristics. Dashes indicate insufficient representation in the literature to judge. Algorithm specific notes are as follows:
*AIM varies with implementation. The standard released code is based on abstract ICA features, but its explanatory ability can be improved and reliance on training data decreased by using parametric filter kernels such as log-Gabor or Difference of Gaussian filters.
**BBM is independent of the visual information, and is really not intended to be applied directly to an image. Rather, it was intended to make a point about motor influences in fixation location, and is designed to be used in conjunction with a visual saliency algorithm.

a different subset of models using a specific hardware setup. Therefore, in order to ensure the fairest comparison over the widest set of models possible, a speed test was conducted using all models currently available in the Saliency Model Implementation Library for Experimental Research (SMILER, discussed in detail in Chapter 3). It should be noted that SMILER does include some small additional computational overhead, but this overhead is consistent across models and should therefore not affect the relative rankings of models.

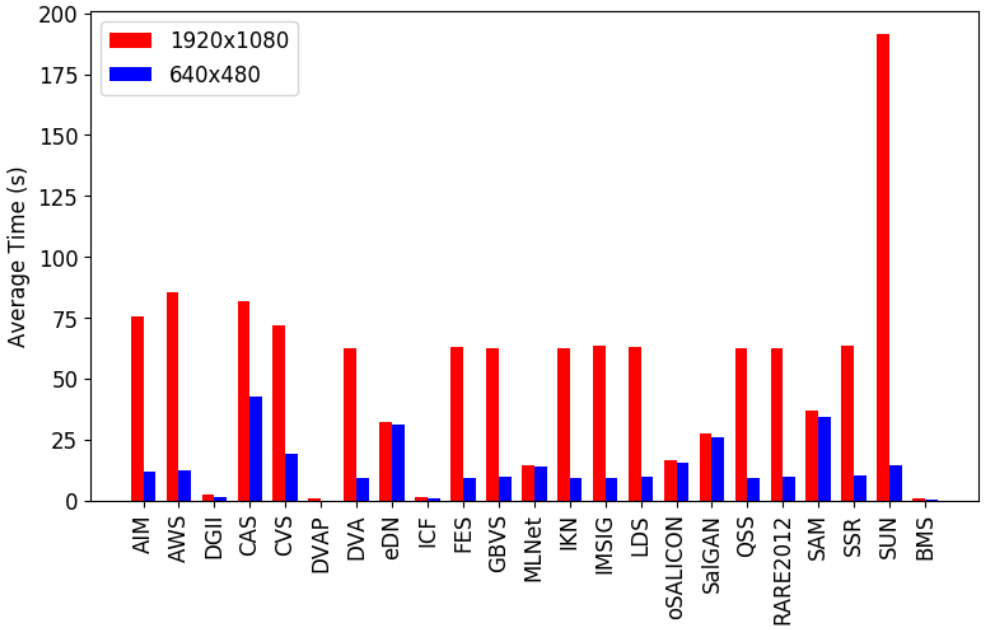


Figure 1.4: Average execution time needed to produce a saliency map for input images of size 1920×1080 and 640×480 pixels for all models currently available in SMILER.

The speed test was conducted on a computer with the following specifications: AMD Ryzen 5 1600X Six-Core 64-bit Processor, 16GB RAM, and NVIDIA GeForce GTX 1080 PCIe 3.0 8GB graphics card. The average time necessary to compute a saliency map was calculated for input images of size 1920 and 640×480 pixels. Results are shown in Figure 1.4. It is important to note that the use of the MATLAB programming language for many of these models may unfairly skew these results; although often a popular choice for prototyping and algorithmic development, models implemented in MATLAB consistently perform more slowly than those implemented in other languages for which execution speed is more optimized. Likewise, a number of the models based on deep learning would either not run or would run much slower without access to a GPU

of sufficient power.

Thus, while it is recommended that any application of saliency should test a number of different algorithms to find the best suited method, the overall trend in algorithms will likely tend toward the follow selections:

- Tasks for which performance in a specific domain is most important and for which there is sufficient training data should probably use a CNN-based approach
- Tasks in which the type of visual input is either relatively unknown or highly variable might be best-served by information theoretic approaches, possibly with some application-specific optimization of parameters and post-processing
- Tasks for which processing speed is paramount on limited hardware may be best served by spectral approaches or models like BMS, that can be implemented to provide fast results in a computationally efficient manner
- Prescribed feature models may be of interest for interdisciplinary research in which specific predictions are being made not just of what is salient but also of how that decision is computed

Of course, this is not an exhaustive list of possible situations for applying saliency, and the challenge of testing and selecting an appropriate model for a new application area is a primary motivation for the developments presented in Chapter 3. Nevertheless, it will hopefully provide some guidance as a starting point for those interested in incorporating saliency methods into their own application area.

1.3 Organization of Dissertation

The remainder of this document is organized along three lines of investigation into the role of the saliency map in fixation prediction and early visual attention. Chapter 2 provides an overview and critical appraisal of metrics used to evaluate how well saliency maps predict human fixations during free-viewing. In particular, the issue of centre bias is analyzed in detail, and a novel approach to evaluation using spatial binning of fixation data is proposed and demonstrated. Chapter 3 presents the Saliency Model Implementation Library for Experimental Research (SMILER), a software library designed to bring saliency model implementations under a common usage format and

streamline future research efforts in saliency modelling and benchmarking. Sections 4.1-4.3 provide examples of SMILER use, investigating the performance of a number of saliency models in the visual search domain of singleton search and examining spatiotemporal patterns of correlation between human fixations and saliency map values. Chapter 5 introduces a fixation control model (STAR-FC) that incorporates saliency as a component in a larger control network, generating explicit sequences of predicted fixations instead of a static probabilistic saliency map. STAR-FC not only produces fixation sequences that are qualitatively much more natural than a rank-order traversal of a saliency map, but also quantitatively much more similar to human fixations when analyzed with a number of common trajectory comparison metrics. Conclusions and future directions of work are provided in Chapter 6.

Chapter 2

Metrics and the Centre Bias

The work in this chapter has been published previously as the following:

Calden Wloka and John K. Tsotsos, "Spatially Binned ROC: A Comprehensive Saliency Metric", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June, 2016

and has been updated and extended here for the purposes of this document.

2.1 Motivation

A continuing challenge in saliency modelling is the formulation of fair and informative metrics with which to evaluate and compare different saliency algorithms. The problems and choices involved in this task are myriad; as mentioned in Section 1.2.2, even electing to focus evaluation on fixation prediction in free-viewing is a specific choice that is by no means conclusively the best option. However, even within that specific domain, there are a number of choices and parameters that may not be immediately obvious but which still might have a significant impact on the results obtained. For example, the choice of images for inclusion in the dataset used for testing, eye tracking methods, or data collection protocol can all have a profound effect on the calculated performance of models. Additionally, there are issues of scale, saliency map numerical spread, and potential confounds within the fixation data itself that must be contended with (see Bruce *et al.* [157] for a comprehensive discussion of many of these issues). Nevertheless, there is an understandable desire to quantitatively compare different models, which has led over the years to a number of metrics adapted from signal analysis or developed explicitly for measuring saliency performance.

Given the focus of modern saliency benchmarking on human free-viewing eye tracking data, the majority of the discussion in this chapter concerns metrics that are based on such data as the ground-truth against which saliency algorithm performance is measured and the manner in which those metrics have failed to adequately deal with spatial biases inherent in the human data. However, as noted in Section 1.2.2, the functional role of saliency is not necessarily settled, and not every application of saliency is likely to be best evaluated by the same metric. Therefore, a brief discussion of alternative metrics and applications is given in Section 2.2.3. Additionally, it should be noted that virtually all saliency algorithm metrics concentrate in some manner on the performance of the resultant map, but there may be additional considerations, such as processing time and resources (or resources saved by the application of a saliency map), which might also be important to consider when selecting an algorithm for a particular task.

2.1.1 Centre Bias in Fixations

Early in the study of saliency it was noticed that stimulus location had a strong effect on the likelihood of fixation, with regions closer to the image centre being more commonly fixated than

those near the image edge [96]. This topic was revisited by Zhang *et al.* [159], who discussed in detail the confounding effect centre bias can have on several popular metrics used to evaluate saliency algorithms (area under the curve (AUC) of a receiver-operator characteristic (ROC) curve discussed further in Section 2.2.2, and the Kullback-Leibler (KL) divergence discussed further in Section 2.2.1). As they pointed out, a saliency map consisting solely of a centered Gaussian outperformed many of the leading saliency models in predicting human fixations despite being independent of the actual image content. Likewise, particularly given the small image sizes being tested, differences in the thickness of the border region left undefined by filter convolution had a tendency to reward models with a greater undefined border due to a concentration of saliency values toward the image centre. While acknowledging that *photographer bias* (the tendency to centre objects of interest in an image) may mean that image centres may genuinely be more likely to be salient than the peripheral locations, they nevertheless approached the issue of centre bias as a problem that had to be solved, implicitly making the assumption (which has since been heavily propagated through the saliency literature) that it is possible to isolate the intrinsic salience of visual stimuli from their spatial context. The assumption that centre bias is both correctable and that it is desirable to perform this correction is potentially rather problematic, however.

While it is, of course, disappointing to have a static Gaussian centre prior outperform one's algorithm in predicting fixation locations using traditional metrics, this does not necessarily mean that such metrics are wrong. Much of the issue at the heart of the debate over metrics seems to rest with an unclear definition of their goals [192]. If the motivation of a model is in producing the best possible predictor of human fixation locations in an image (eg. when applying saliency algorithms as a processing step in image compression), then it does not particularly matter whether a correct pixel label is based on a positional prior or the visual content of the image. In fact, significant evidence exists that there is an inherent bias to human fixations toward the image centre [170, 193]. The root of this bias, however, is not completely understood; it could simply be a physiological tendency to keep the eye in a neutral position, a strategic choice to maximize visual foraging [114, 115, 193], the emergent statistics of saccadic eye movements [194], or some combination of all three. Regardless of the source, there is no guarantee that fixation spatial bias will interact linearly with saliency, and thus efforts to isolate saliency map generation from the spatial distribution of eye movements may be artificially restricting the capacity of saliency models to accurately predict

human fixation patterns.

Likewise, while some centre bias may be created by photographer bias toward centering objects of interest in a frame, this should actually have very little effect on algorithm performance in classical metrics once image border effects are controlled for. After all, if the most interesting visual stimuli consistently appear near the image centre then a high performing saliency algorithm should likewise consistently detect the image centre as most salient. This may be empirically explored by examining the eye tracking data from two different data sets: the **D**atabase **O**f **V**isual **E**ye Movement**S** (DOVES) [5], and the MIT dataset of human eye-tracking [4].

It is important to note that eye fixations in both the MIT and DOVES datasets were captured during free-viewing. It has long been established that task can have a profound effect on fixation patterns; this was first suggested by the seminal work of Yarbus [10] and recently explored more systematically by Borji and Itti [195]. Although some saliency work has attempted to incorporate task bias [196, 197], the majority of saliency modelling is nevertheless done under the assumption of free-viewing.

Statistical properties of the free-viewing fixation patterns for human observers of these datasets are presented in Table 2.1. All values have been normalized with respect to the image dimensions, and therefore, although the proportional variance of the DOVES fixations is nearly identical in both the x- and y-directions (0.14 and 0.13, respectively), the fixations along the x-direction actually do have a greater spread in terms of raw pixel distances. The MIT dataset, available at [198], is composed of 1003 images sampled from Flickr creative commons and LabelMe [199] with eye tracking data for fifteen observers. Although it is never possible to have a completely representative dataset of images, the MIT set provides a decent attempt to capture a cross-section of the types of photographs people take and share with others (*e.g.* Figure 2.1). Human fixations over this dataset are strongly biased toward the image centre; at least a portion of this bias likely arises due to photographic composition. In order to compile distribution statistics shown in Table 2.1 for the MIT dataset, which includes images of different dimensions, analysis was limited to only those 463 that were 1024×768 pixels (landscape) and 123 that were 768×1024 pixels (portrait) in size (the most common sizes in the set).

The DOVES dataset, available at [200], consists of 101 grayscale images cropped from the dataset originally created by [201]. All images in the DOVES set are of landscape orientation with

dimensions 1024×768 pixels. In contrast to the MIT dataset, the DOVES dataset provides a strong attempt to mitigate any bias inherent in photographic composition. Images that contained man-made structures, faces, animals, or other objects deemed to be of obvious semantic interest were explicitly omitted, and the additional step of cropping a sub-portion helped to disrupt compositional framing. Subsequently, images in the dataset have no clearly framed central object or creature (*e.g.* Figure 2.2). Despite this lack of compositional bias in the image stimuli, aggregate fixation statistics over the dataset shown in Table 2.1 display that human fixations remain distinctly biased toward the image centre (albeit to a lesser extent than in the MIT dataset). Given the lack of strong central objects, this centrally biased distribution pattern most likely corresponds to factors independent of the visual qualities of the stimulus.

Figure 2.1: A randomly selected example image from the MIT dataset [4]. As with many of the images, there is a strong central subject with little peripheral content.

We can formulate a spatial prior for eye fixations in the following manner: At the most basic level of abstraction we consider eye fixation data over a visual field as a sequence of points constrained to the 2D plane of the image. Without any knowledge of the underlying visual stimulus (given that we are formulating a prior), an initial best guess for a fixation will be a drawn from a random distribution $p(x, y)$, where p is the probability distribution and (x, y) are the current pixel coordinates of gaze. Each subsequent fixation is dependent only on the previous location in the

Figure 2.2: A randomly selected example image from the DOVES dataset [5]. As can be seen, there is no central object of interest.

chain (for now ignoring, for the sake of simplicity, inhibition of return), and thus the t-th fixation takes the form

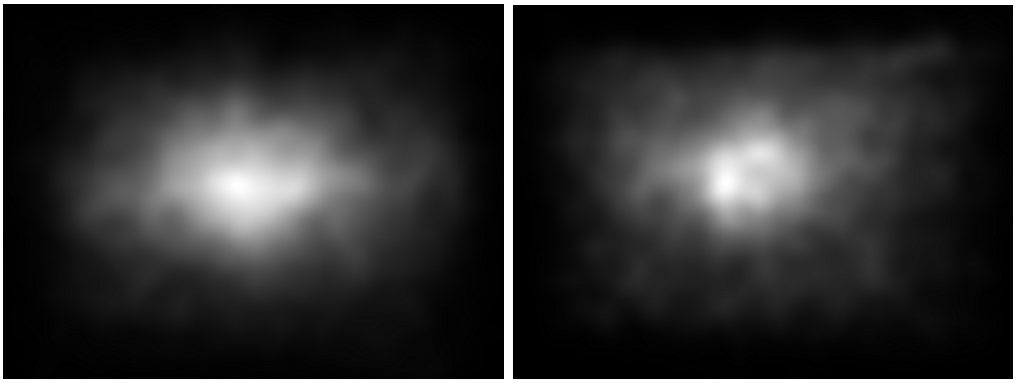
$$(x_t, y_t) \sim p(x_t, y_t | x_{t-1}, y_{t-1}) \tag{2.1}$$

which is the definition of a random walk.

Each specific image in a dataset corresponds to a single independent sampling of the random walk. As one would expect by the Central Limit Theorem, it can be shown that the distribution of the point conglomerate produced by this process will tend toward that of a Gaussian distribution

		Mean	Variance
DOVES Database	x	0.00	0.14
	y	0.00	0.13
MIT Database Portrait	x	0.00	0.027
	y	0.00	0.040
MIT Database Landscape	x	0.00	0.035
	y	0.00	0.028

Table 2.1: Distribution statistics over all human fixations collected in psychophysical eye-tracking datasets. Values are normalized with respect to image dimensions, and show that fixations consistently cluster around the centre of the image (0 mean) rather than off-centre, but range in variance. Thus, while degree of bias varies, the bias itself remains consistent.



(a) MIT Landscape **(b)** DOVES

Figure 2.3: Fixation cloud images formed by smoothing over all human fixations in the dataset. On the left is shown the fixation cloud for landscape-oriented images in the MIT dataset, and on the right the fixation cloud for the DOVES dataset.

[202]. Empirically, we demonstrate this in one dimension by generating random walk trials with sequences of five fixations over a uniform subinterval of the normalized domain $[-1, 1]$. The approximate distribution for this fixation set is formed from the smoothed histogram of the fixation locations. Figure 2.4 shows how after 1000 trials this approximate probability distribution very closely matches a Normal distribution of identical variance.

Thus, we see that the Gaussian central prior, which is a prevalent prior used in a number of models and metrics, can be derived by a simple translating saccadic model. Additional efforts to model the dynamic process of saccadic eye movements with a random walk includes both Brockman and Geisel's [203] and Boccignone and Ferraro's [204] work showing that saccadic movements can be well captured as stochastic sequences over a saliency field that correspond well to Lévy flight random walks. Therefore, rather than a confounding artifact that must be corrected for, the centre bias of human fixations can be seen to derive, at least in part, from the mechanics of how people look. Likewise, the improvement in fixation prediction seen by the addition of a Gaussian centre prior is due to the fact that a Gaussian functions as a first-order approximation to the actual spatial biases that are introduced through active gaze mechanics. As a result, a model of saliency should inherently account for these effects rather than view their manifestation as a nuisance that must be separately corrected for. At the very least, metrics that are used to evaluate this performance should not obscure or discount this component of human fixation selection. Of course, we still seek a fair method of evaluating human fixation prediction for algorithms with varying degrees of

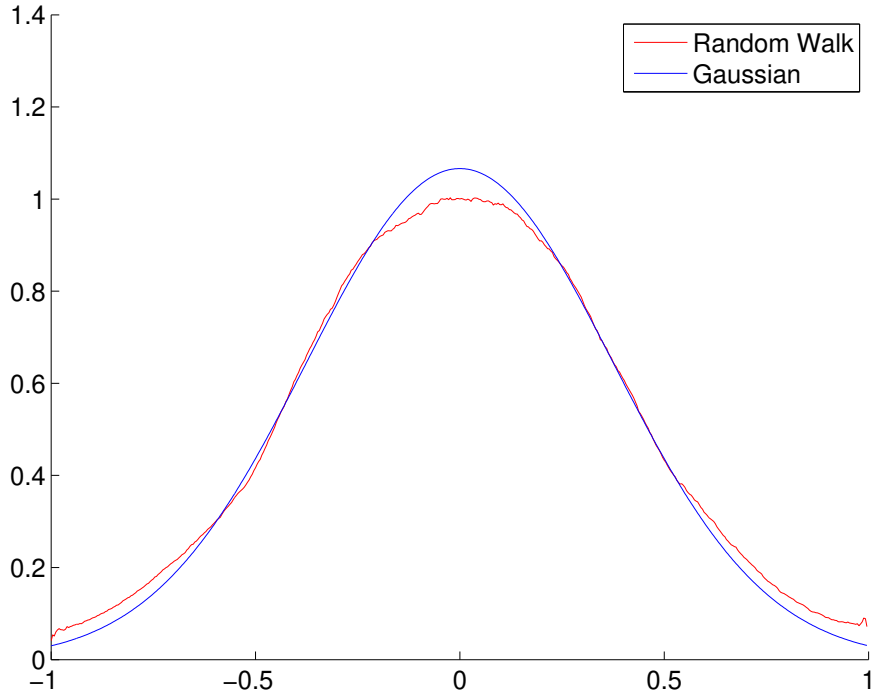


Figure 2.4: A comparison of the approximate distribution curve for fixations produced by a random walk plotted against a Gaussian of identical variance.

spatial bias representation, and would like this metric to represent algorithm performance across the entire image rather than have the measure of performance be overwhelmed by the central signal. Section 2.3 presents a proposed solution to the issue of both clearly and fairly evaluating algorithm performance.

2.2 Review of Saliency Scoring Metrics

Before presenting a novel evaluation metric, it is useful to review in some detail the metrics that have already achieved wide usage in evaluating saliency algorithm performance.

2.2.1 Value-Based and Distribution-Based Metrics

Value-based metrics are evaluative techniques that take into account the actual saliency scores assigned to pixel locations rather than simply the rank order of assignment. Distribution-based metrics use statistical approaches to compare the overall distribution of salience values in a saliency map to the ground-truth provided by human fixation.

2.2.1.1 Normalized Scanpath Saliency (NSS)

Normalized Scanpath Saliency (NSS) is a value-based metric developed specifically for saliency algorithm evaluation first proposed by Peters et al. [205]. The metric starts by normalizing the saliency map for a given image such that the map has zero mean and unit standard deviation. The score is then calculated as follows:

$$\mathbf{NSS} = \frac{1}{N} \sum_{f \in \mathbf{F}} S_N(f_x, f_y), \tag{2.2}$$

where $\mathbf{F}$ is the set of human fixations, $f = (f_x, f_y)$ are the (x, y) image coordinates of a given fixation, N is the total number of human fixations in the ground-truth set, and $S_N(x, y)$ is the normalized saliency value in saliency map location (x, y). By initially normalizing the map to have zero mean, NSS scores can be quickly and intuitively assessed: scores above zero suggest the model predicts human fixations above chance, whereas scores below zero suggest that model predictions are anti-correlated with human fixation locations. Also, by normalizing the maps of different saliency algorithms onto the same distribution, NSS scores can be directly compared between algorithms.

Based on its reliance on only a small number of values sampled from each saliency map, NSS is highly vulnerable to noise, makes no attempt to control or correct for centre bias, and can be greatly influenced by the degree of post-processing smoothing [169]. Due to these considerations, NSS has fallen out of favour as a metric in most recent large-scale benchmarking work, and is either neglected completely [1, 206] or recommended against in lieu of other available metrics [169]. Nevertheless, some studies have retained the NSS metric and recommended its use in conjunction with one or two other metrics in order to provide a more robust overall picture of algorithm performance [148].

2.2.1.2 Correlation Coefficient (CC)

The Pearson correlation coefficient (CC) is a measure of the similarity between two maps, given by the equation:

$$\mathbf{CC} = \frac{cov(M_1, M_2)}{\sigma_1 \sigma_2}, \tag{2.3}$$

where $cov(M_1, M_2)$ is the covariance between the first map, M_1 (typically the human fixation locations convolved with a Gaussian kernel), and the second map, M_2 (the saliency map), and σ_1

and σ_2 are the standard deviations of the map values for maps one and two, respectively.

Similar to NSS, CC values greater than zero indicate a positive correlation, whereas CC values below zero indicate anticorrelation. However, the CC measure is additionally constrained to the range of $[-1, 1]$. Because the use of human fixation coordinates as the ground truth tends to create relatively sparse target maps, even after smoothing with a Gaussian kernel, CC measures tend to reward highly sparse algorithms (such as DVA and QSS) in preference to algorithms that produce more fully populated saliency maps (such as IKN and AIM). Additionally, CC does not control for centre bias. Overall it is not a widely used metric, though has been included in some large benchmarking studies (namely, Borji et al.'s work [169]).

Potentially more promising than the Pearson CC is the Spearman rank correlation coefficient. Rather than assessing the linear correlation between two distributions, the Spearman correlation assesses the correlation over the rank ordering of values, and is thus able to assess any monotonic relationship. It is therefore more robust to different dynamic ranges in saliency map values between models. However, a primary challenge in using the Spearman correlation is the ambiguity introduced by tied values. This variant of the correlation coefficient is currently not a standard metric used by major benchmarking efforts (such as the MIT Saliency Benchmark [79]), but it has been introduced in the realm of salient object detection [207] and could potentially be a useful adaptation for fixation prediction evaluation.

2.2.1.3 Earth-Mover's Distance (EMD)

The Earth-Mover's Distance (EMD) gets its name due to an intuitive analogy for its function: if the manifolds for two spatial probability distributions represent piles of dirt (where the dirt is piled higher for regions of greater probability), EMD is the amount of work, defined by the total distance through which dirt has to move, required to turn the profile of one distribution into the other. It was defined in the image processing domain by Rubner et al. [208] and is governed by the following

equation:

$$EMD(P_1, P_2) = \left(\min_{f_{i,j}} \sum_{i,j} f_{i,j} d_{i,j} + \left| \sum_i P_{1i} - \sum_j P_{2j} \right| \overset{\max d_{i,j}}{i,j} \right), \quad (2.4a)$$

$$s.t.\ f_{i,j} \geq 0 \quad \textstyle\sum_j f_{i,j} \leq P_{1i} \quad \sum_i f_{i,j} \leq P_{2j} \ \sum_{i,j} f_{i,j} = \min \left(\sum_i P_{1i}, \sum_j P_{2j} \right), \quad (2.4b)$$

where P_1 and P_2 are two spatial probability distributions, $f_{i,j}$ represents the amount transported from the ith source in P_1 to the jth destination in P_2, and $d_{i,j}$ is the distance between locations i and j in the distribution layouts. The lower the EMD score the closer the two distributions are. Unlike KL divergence (see below), EMD is a true metric. It has not been widely used in the saliency literature, but has seen limited use (see, for example, [1]). Many of the considerations and caveats involved in the use of CC are likewise reflected in EMD. However, EMD is more flexible in its definition of distributions than CC, and allows for comparisons between differently sized and shaped distributions (so long as a distance from one location to another can still be defined). This opens the door to additional forms of post-processing and scale-space processing of saliency maps that would not otherwise be available in CC.

2.2.1.4 Kullback-Leibler (KL) Divergence

The Kullback-Leibler (KL) divergence [209] is used to estimate the dissimilarity between two distributions. It is given by the equation:

$$KL(P_1, P_2) = \sum_{(x,y) \in R} P_1(x, y) \log \left(\frac{P_2(x, y)}{P_1(x, y)} \right), \quad (2.5)$$

where P_1 and P_2 are probability distributions over a neighbourhood of pixels R and (x, y) are the coordinates of a specific pixel in that neighbourhood. In order for the KL divergence to be defined, both P_1 and P_2 must be true probability distributions (sum to 1) and must have non-zero elements such that the output of the logarithm remains defined. The KL divergence varies from 0 (a perfect match) to infinity. It is important to note that the KL divergence is not a true metric, in that it is not symmetric and does not satisfy the triangle inequality. This, combined with its non-linearity and lack of an upper bound, makes interpretation of its scores beyond relative rank

order between algorithms somewhat challenging. However, the asymmetry can also be viewed as a way of privileging one distribution (P_1) as the target or true distribution. Additionally, the measure can be made symmetric by taking the average of both directions (sometimes known as the relative entropy of two distributions):

$$KL(P_1, P_2) = \frac{1}{2} \sum_{(x,y) \in R} \left(P_1(x,y) \log \left(\frac{P_2(x,y)}{P_1(x,y)} \right) + P_2(x,y) \log \left(\frac{P_1(x,y)}{P_2(x,y)} \right) \right). \tag{2.6}$$

There are multiple ways in which the KL divergence has been applied to gauge saliency algorithm performance. For example, Le Meur and Baccino [210] apply the KL divergence of Equation 2.5 to the full field, whereas Itti and Baldi [211] and Itti and Koch [50] compare the distribution of saliency values in a small neighbourhood around human fixation locations against randomly sampled locations in other parts of the image using the relative entropy of Equation 2.6. It is important to note that for Le Meur and Baccino, the closer the score is to zero the better, whereas the interpretation of the latter application is the opposite. In the second version of use, human fixations are not directly compared against, but rather the KL divergence is used to test how differently saliency is defined at fixated points versus non-fixated points (with the assumption that a good saliency algorithm will be strongly different between fixated and non-fixated locations).

2.2.1.5 Information Gain

Rather than viewing the raw saliency maps as distributions for comparison, Kümmerer *et al.* propose an information theoretic formulation that attempts to provide a more consistent and fair ranking of models. The methodology hinges on a number of assumptions and processing steps:

- Reinterpreting saliency maps as a probability field and rescaling across the dataset to maximize log-likelihood
- Calculation of a "baseline" model across a dataset
- Calculation of a "gold standard" model derived from the ability of one human's results to predict another's

To accomplish the first point, all saliency maps for a given model are jointly re-scaled to have values within the range $[0, 1]$ across a given dataset. A Gaussian blur of radius σ is applied, along

with a monotonic non-linearity (such that it does not affect AUC score) in the form of a continuous piecewise linear function formed by twenty evenly spaced segments over the range of possible salient values, and finally a centre bias term formed by a twelve segment piecewise linear function. All transform parameters are jointly optimized for each model.

This approach has a number of advantages. By optimizing all models for spatial bias and blur in a similar manner, the ordering of performance becomes consistent across metrics. The use of information gain as a metric, which is measured in bits, also allows for more intuitive interpretations of scores in comparison to metrics such as AUC (which saturates, and thus numerical improvements of equivalent magnitude at different ranges of performance are not necessarily equivalent in terms of degree of improvement in model performance).

However, while optimizing independent parameters for blur and spatial bias allows for more consistency of evaluation rankings, it also makes the assumption that these parameters are independent from the performance of the saliency algorithm and, more critically, the task of calculating a saliency map. As argued in Section 2.1.1, this may not be a valid assumption, and obscures the role that these optimizations play in determining the resultant evaluation of a model's performance. By optimizing in a way that is dependent on the traits of the dataset itself, it makes it unclear what parameters are best used if deploying a model in a role for which these dataset optimizations are not available (for example, in a live environment or over a set of images lacking in human ground-truth).

2.2.2 Binary Methods

Binary methods treat saliency algorithms as classifiers designed to label pixels as either salient or not in a binary fashion. They tend to be largely invariant to a number of underlying algorithm characteristics and are well-established signal detection methods, making them a popular class of metric. Due to the thresholded transformation of maps into a binary classifier, these techniques are invariant to any monotonically increasing transformation applied to the saliency values themselves, facilitating direct comparisons between algorithms.

2.2.2.1 Precision-Recall Curve

A Precision-Recall curve is constructed by sweeping through a saliency map with a binary threshold, and at each threshold calculating the number of *true positives* (TP), *false positives* (FP), and *false negatives* (FN) in order to calculate the *precision* and *recall* values according to the equations:

$$Precision = \frac{TP}{TP + FP}, \tag{2.7}$$

$$Recall = \frac{TP}{TP + FN}. \tag{2.8}$$

Because the number of fixated pixels in an image is usually much lower than the total number of pixels, what constitutes the positive and negative set will sometimes be redefined in order to make both sets closer in size. For example, in Borji et al.'s benchmark work on salient object detection [206], the positive set is defined by full bounding boxes around the target objects, greatly increasing the pixel count of the positive set. In the majority of the saliency literature, however, the precision-recall curve is omitted and instead the related concept of the Receiver Operating Characteristic (ROC) curve is used.

2.2.2.2 Receiver Operating Characteristic (ROC) and the Area Under the Curve (AUC)

Receiver Operating Characteristic (ROC) curves have a long history of use in psychophysical analysis [212]. An ROC curve plots the true positive rate (which is the number of detected true positives divided by the total number of true positives) against the false positive rate (which is the number of selected negative elements divided by the total negative set). Because positives and negatives are both normalized by the total set size, the axes of ROC curves will always be over the range $[0, 1]$ regardless of the relative sizes of the two sets. Additionally, the area under the curve (AUC) provides a compact representation of the performance curve. AUC values will always fall on the range $[0, 1]$, with scores above 0.5 representing better-than-chance performance.

Despite the seeming straightforward elegance of the ROC metric, there has been frequent disagreement and revision over how best to define the positive and negative sets, leading to a number

of different versions of the saliency ROC. The initial definition, pioneered by Bruce and Tsotsos [2] based on the work of Tatler *et al.* [95], defined the positive set to be all fixated pixels and the negative set to be all other pixels in the image. However, it was rather quickly noticed that centre bias is a significant factor in AUC scores [159], which has sparked a number of competing proposals on how to modify the ROC metric to address this confound.

Judd et al. [1] retain the original formulation of positive and negative sets (hereafter referred to as *classical* ROC, or *c*ROC), but do an individually optimized weighted convolution of each algorithm's saliency maps with a Gaussian centre prior. The justification is that the *c*ROC metric provides the truest measure of raw performance, and by optimizing each algorithm to take advantage of a spatial prior the comparison between algorithms both becomes fair and provides a measure of the maximally achievable predictive performance for each algorithm. However, rather than solving the problem, this approach confounds the metric by incorporating the same bias as exists in the data, and it becomes difficult to judge whether a model's performance is based on its predictive capabilities or the optimized prior.

Li et al. [60] largely retain the *c*ROC structure, but instead of convolving each algorithm's output with a central prior, they enlarge the border of undefined values caused by convolution with the algorithm's feature kernels such that all algorithms have an equivalent border of zero-saliency pixels. Although this may seem like a minor consideration, Zhang et al. [159] previously demonstrated that even this relatively minor change can have a significant impact on AUC score. Nevertheless, such an approach would not penalize static maps (*e.g.* a centered Gaussian map) that are independent of the underlying image, and was therefore viewed as insufficient for dealing with the issue of centre bias.

The most radical adjustment to the definition of the ROC metric components, however, was proposed by Zhang et al. [159] based on the work of Tatler *et al.* [95] and Parkhurst and Niebur [213] in the form of the *shuffled* ROC (*s*ROC). In *s*ROC, the positive set is still taken to be the human fixation points for a given image, but the negative set is redefined to consist of a random sample of human fixations from other images in the same dataset. By redefining the negative set in this way, any spatial bias or static prior that makes it more likely to select elements from the positive set will have an approximately equal probability to likewise increase the false positive rate. This property has made *s*ROC highly popular in recent contributions to the saliency literature,

both for large-scale benchmarking [169, 148] and as the metric of choice for demonstrating the efficacy of more recently proposed algorithms [150, 175].

Nevertheless, despite this widespread adoption, there are a number of caveats worth pointing out regarding sROC. Unlike with cROC, the AUC score resulting from the shuffled metric lacks an easily interpretable physical interpretation. In cROC methods, the performance curve can be understood as a direct measure of the likelihood of successfully predicting a human fixation point at a given cut-off threshold. The sROC curve, however, does not share this straightforward interpretation due to its stochastic nature and lack of transparency of how many genuinely salient targets were likewise counted as false positives due to their spatial location. While this does not affect the utility of sROC in a relative comparison of algorithm performance, it does make it difficult to interpret the actual meaning of the numerical results. In particular, there are a small number of algorithms that actually score better when measured by sROC than with cROC [157], which suggests that sROC may actually be introducing its own set of difficult to detect biases.

Of course, while it is perhaps disappointing to have a static Gaussian centre prior outperform one's algorithm in predicting fixation locations using traditional metrics, this does not necessarily mean that such metrics are wrong. Much of the debate over metrics seems to rest with an unclear definition of their goals [192]. If the motivation of a model is in producing the best possible predictor of human fixation locations in an image (*e.g.* for use in image compression), then it does not particularly matter whether a correct pixel label is based on a positional prior or the visual content of the image. The shuffled ROC also makes a fundamental assumption that it is actually possible to isolate the intrinsic salience of visual stimuli from its spatial context that, as discussed in Section 2.1.1, may not be a valid premise.

2.2.3 Future Directions for Performance Metrics

All of the previously discussed metrics are predominantly applied to saliency maps using a static ground-truth set formed by human fixations on the same image. However, there are a number of additional avenues of prediction along which it would be natural to measure saliency algorithm performance. Much as the complexity of object recognition benchmarks and challenges has increased over the years as the field matures and previous tests, metrics, and benchmarks are exhausted [214, 215, 216], so too it is likely that saliency algorithm research will need to expand its scope

to include new challenges, problems, and measures of performance. This expansion of evaluation methods will be particularly important as the manner in which saliency algorithms find practical applications increases (see, for example, Section 1.2.4) in order to ensure that one truly is selecting the most appropriate tool for a given task.

Some efforts at defining new application-specific benchmarks and metrics has already begun. Both Feichtenhofer *et al.* [138] and Roberts *et al.* [29] define application-specific notions of saliency for their specific problem domains (action recognition and robot navigation, respectively), demonstrating the effectiveness of these solutions based on performance comparisons on established benchmarks both with and without the saliency component. Nevertheless, there appears to be little work in comparing the effect on performance of the saliency component itself to alternative methods of computing saliency, suggesting that this is a ripe area for investigation. In fact, Zhang *et al.* [217] recently released such a study for the domain of image quality assessment (IQA). It is likely that saliency is a concept that is general enough that this kind of application specific testing will become increasingly important, as there is no guarantee that what correlates well with performance in one application domain will translate to another.

In addition to finding new, possibly application-specific ways of evaluation, however, it will be an ongoing challenge to ensure that evaluation methods are as clear and methodologically sound as possible. For example, gaze prediction has been a focus of saliency performance testing for long enough that a number of methodological criticisms and challenges have been identified [157], but these points do not appear to have been considered by Zhang *et al.* [217] when evaluating IQA, as they do not appear to make any attempt to control for the influence of centre bias, post-processing smoothing, or range of saliency map values. Both Borji *et al.* [125] and Riche *et al.* [148] conclude that a robust evaluation of model performance is best obtained by combining complementary metrics, with the ROC class of metrics as a frequent focal point of algorithm analysis. Nevertheless, the central bias in human fixations has motivated the use of *s*AUC as standard practice. However, as outlined above, *s*AUC has a number of methodological issues and centre bias is better understood as an intrinsic aspect of active foveal vision for which it may not be possible to work around.

Motivated by the need for a clear, intuitively understandable metric, a novel evaluation metric is proposed in Section 2.3. This metric analyzes saliency algorithm prediction of human fixations

within the context of their spatial distribution over the dataset by spatially binning the ground-truth fixation points and then deriving an ROC curve for each bin independently.

2.3 Spatially Binned ROC

The spatially binned ROC (spROC) metric seeks to preserve a useful degree of spatial information while still yielding a clear evaluation of saliency algorithm performance. The metric is constructed in the following manner:

1. Partition the image into a set of non-overlapping spatial regions (bins). Each bin is an annulus (except the central bin, which is an ellipse, and the final outer bin) centered on the image centre[1]. Because of the tendency for human fixations to vary in proportion to the height and width of the image, bin dimensions are determined by the aspect ratio of the image (see Figure 2.5 for examples)

2. For a given image, determine into which bin each ground-truth human fixation falls

3. Calculate a traditional ROC curve for each bin

The selection of the most appropriate size, shape, and number of the spatial bins may be application specific (for example, when viewing web pages viewers will frequently exhibit an F-shaped bias localized in the upper left of the page, rather than a centre bias [218]). We elected to use ten bins and allocate the bins such that each bin had an equal portion of the total set of human fixations (see Figure 2.5). To ease comparison among methods, such a configuration might be considered as the 'standard' one. However, it is possible for some specific applications that one may wish to investigate the performance of an algorithm according to an alternative distribution of bins that is independent of fixational set, such as one that is determined by relative image area or informed by physical considerations such as degree of visual angle forming the thickness of each ring.

One of the advantages of the spROC method is that algorithm performance can be analyzed at a number of levels. The traditional ROC curve can be straightforwardly calculated by taking the

[1]For datasets which are not centre biased, an alternative bin distribution would be required. Ideally, this distribution should capture an approximately equal number of fixations in each bin area while also retaining general semantic interpretations of bin locations.

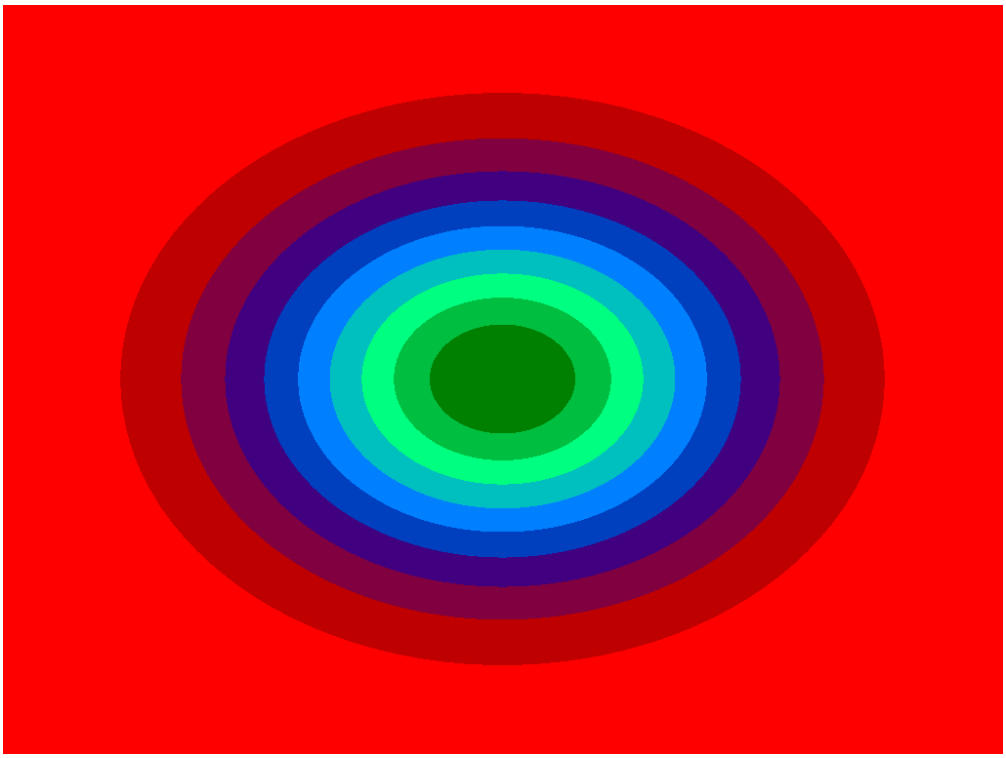


Figure 2.5: Example showing bins distributed on a 4:3 aspect ratio image proportional to the number of fixations from the MIT dataset falling into each bin. Each band of colour represents a different spatial bin.

weighted sum of the individual spatial bins according to the equation:

$$\mathrm{PR}_j = \sum_{i=1}^{n} c_i \mathrm{PR}_{ij} \tag{2.9}$$

where PR_j is the positive rate at threshold j, c_i is the count of fixations falling into bin i, and PR_{ij} is the positive rate in the ith bin at threshold j. Likewise, the traditional AUC score can be calculated by finding the area under this curve. When using a proportional distribution of bins Equation 2.9 simplifies to the average across all bins.

Alternatively, however, one can also examine a spatial profile of the algorithm performance by plotting the AUC score for each individual spatial bin (see Figure 2.6). Algorithms with a spatial bias will exhibit deviations from a horizontal line, and the degree of deviation can be used to quantify the extent of bias. An unbiased algorithm will form a flat line (every bin will have the same AUC score), while a well-performing algorithm will have the best combined score across all bins. It depends on the application which is more important; although a highly biased algorithm

might end up giving the best overall score, the spatial bias exhibited suggests that at least part of its performance is based on an overemphasis (either implicitly or explicitly) on the spatial tendencies of human fixations (the ability to predict less frequent peripheral fixation is sacrificed to improve the chances of predicting central fixations). Adjustment or 'correction' for the centre bias of human fixations can be performed through a re-weighting of the ROC points or AUC score between the bins. This will have an effect similar in outcome to shuffled ROC, but with the added transparency of knowing precisely how fixations have been re-weighted rather than relying on a hidden stochastic process. An example of this type of analysis is shown in the comparison of Tables 2.2 and 2.3, where Table 2.2 shows results using classical AUC, and Table 2.3 displays instead AUC scores weighted by the image area covered by each bin[2].

2.4 Results of Analysis Using spROC

Here we present the quantitative clarity achieved by using spROC for a selection of algorithms that have publicly available code. All algorithms have been run without the application of post-processing smoothing (also referred to as blurring). Although smoothing is a standard practice and is well-known to have a strong effect on the performance of an algorithm's fixation prediction, convolution will introduce an additional bias against peripheral saliency values proportional to the size of the Gaussian kernel used to perform the smoothing. Since algorithms will frequently exhibit different optimal sizes of smoothing kernel (*e.g.* see [1]), we felt it was useful to look at the inherent degrees of algorithm spatial bias that exists prior to applying any post-processing smoothing. Note, however, that while post-processing smoothing was removed, some algorithms still implicitly smooth their output through image resizing. This step is required for efficient processing speed (*e.g.* GBVS) and thus was retained, but does generally lead to improved scores for these algorithms versus those that have no built-in smoothing. Therefore, it is important to reiterate that the scores presented here are not an optimized benchmark (as in [1, 125, 148]), but rather serve as a baseline characterization of the inherent spatial bias for each algorithm.

We demonstrate the spROC metric using the following algorithms not based on deep-learning:

[2]The particular choice in region weighting is best determined specifically for a particular task or application. For example, weighting by image area may prove useful in determining algorithm suitability for anisotropic compression, an application in which the benefits of accurate prediction increase with increasing image area.

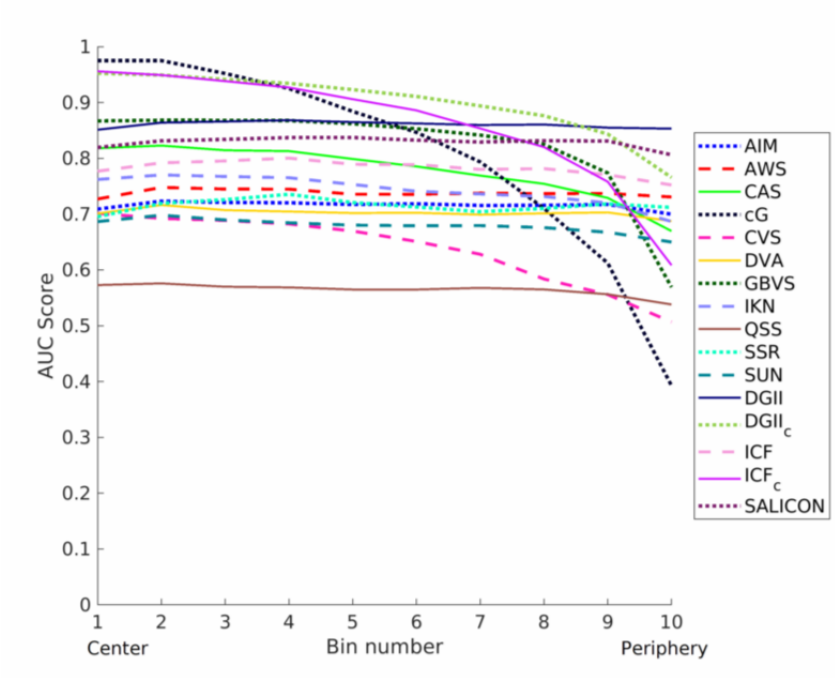
- Attention by Information Maximization (AIM) [2]
- Adaptive Whitening Saliency (AWS) [162]
- Context Aware Saliency (CAS) [143]
- A centered Gaussian prior (cG)
- Covariance-based Saliency (CVS) [145]
- Dynamic Visual Attention (DVA) [160]
- Graph-Based Visual Saliency (GBVS) [78]
- The Itti-Koch-Niebur Saliency Model (IKN) [25]
- Quaternion-Based Spectral Saliency (QSS) [167]
- Saliency Detection by Self-Resemblance (SSR) [161]
- Saliency Using Natural statistics (SUN) [159]

and the following algorithms based on deep learning:

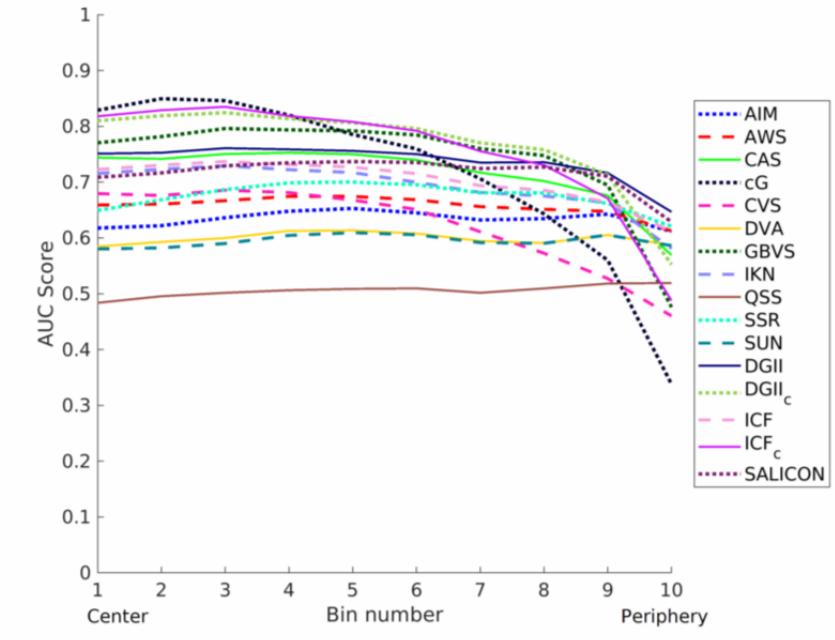
- Deep Gaze II (DGII) [182]
- Intensity Contrast Features (ICF) model [184]
- SALICON [7]

that we ran on two widely used benchmarking datasets: the MIT dataset already discussed in Section 2.1.1, and the ImgSal dataset [60], which was the basis of the benchmarking work by Riche *et al.* [148]. Note that we used the implementation of CAS created by Tsai and Chang [219] to ensure control over post-processing, as the original study authors released only a binary implementation. Likewise, the SALICON implementation used was OpenSALICON [220], as at the time of this writing the original model authors have not publicly released their code. For the DGII and ICF models, they default to including a centre bias component learned from the MIT dataset. However, this centre prior is applied independently from the feature-based map, and so to see how much this effects the results of the model they were run in a mode that includes the centre bias (DGII_C and ICF_C) and in a mode without the application of a centre bias (DGII and ICF).

AUC scores plotted by bin number are shown in Figures 2.6-2.8. Figure 2.6 shows results for all of the tested models over the MIT dataset (Figure 2.6a) and over the ImgSal dataset (Figure 2.6b). Figures 2.7 and 2.8 repeat this information, but the models are split into the set of classic models

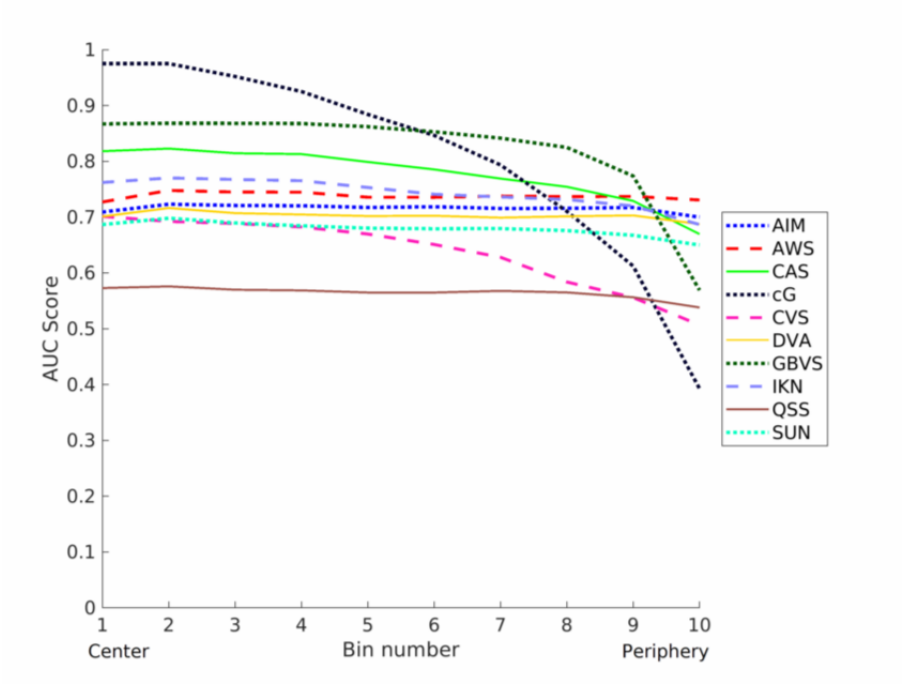


(a) MIT

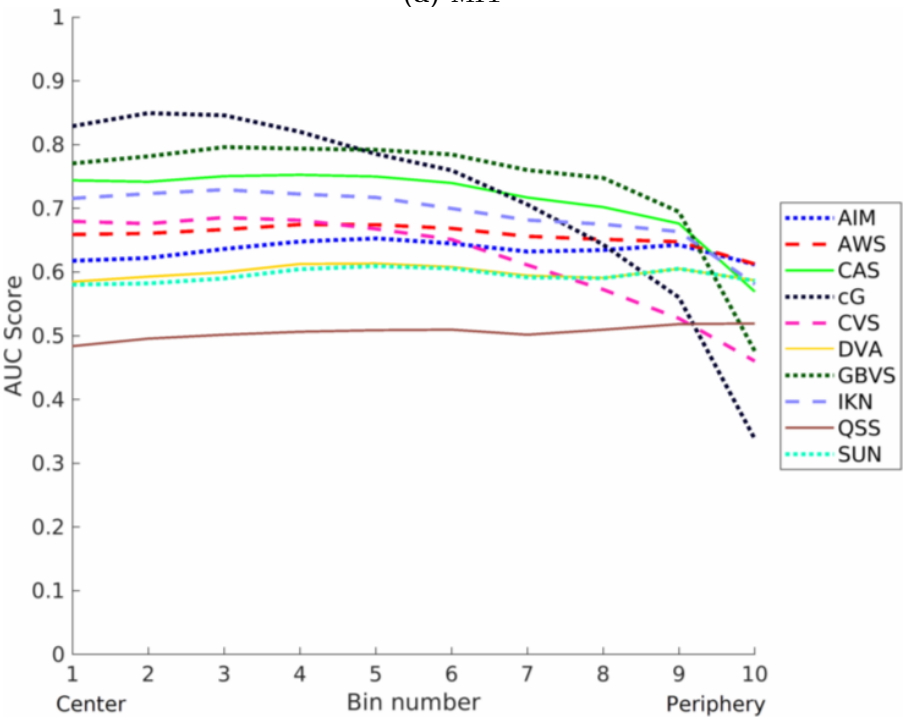


(b) ImgSal

Figure 2.6: spAUC scores for all tested algorithms. (a) presents results over the MIT dataset, and (b) presents results over the ImgSal dataset. All algorithm saliency maps were unsmoothed.

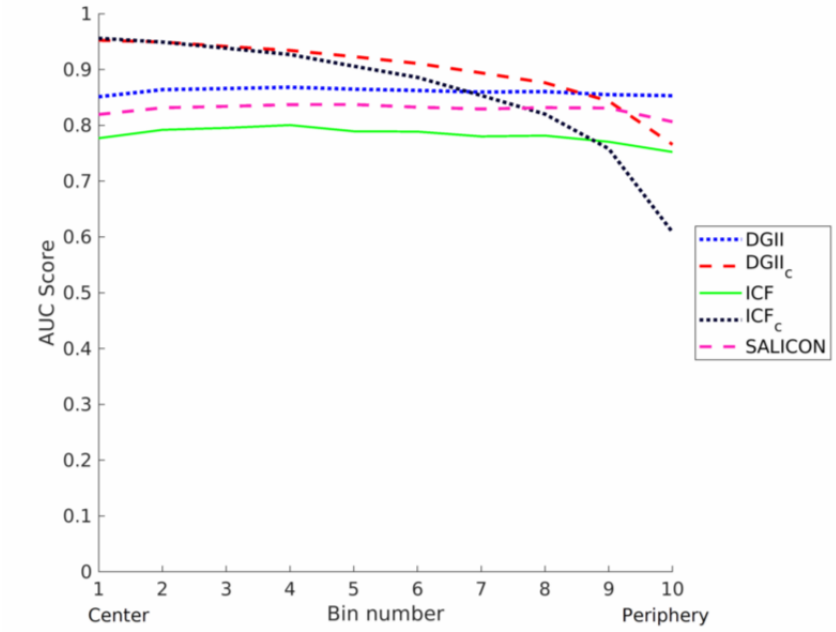


(a) MIT

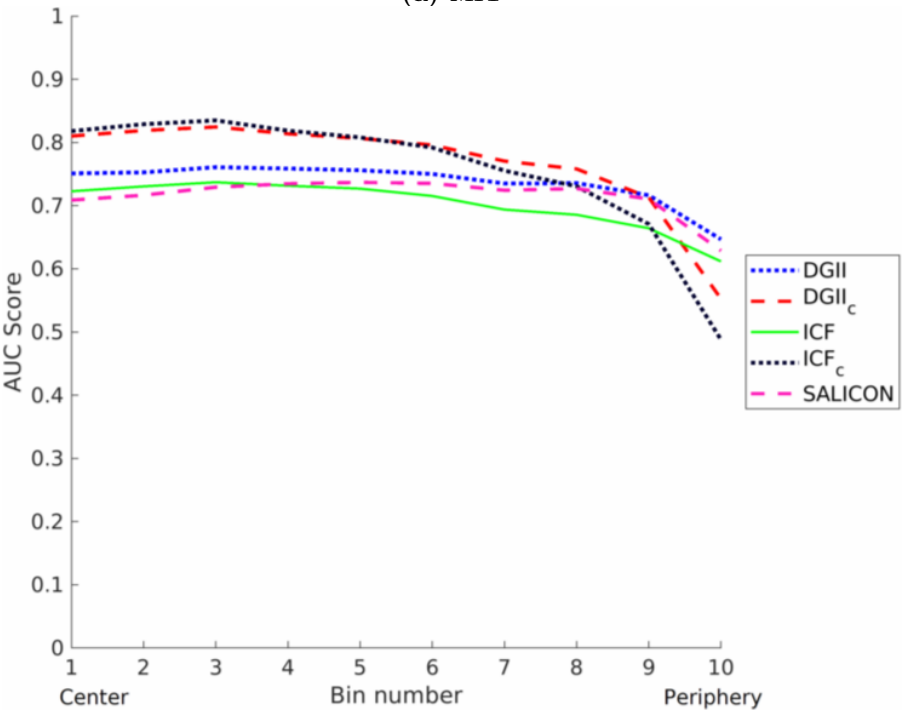


(b) ImgSal

Figure 2.7: spAUC scores for all algorithms tested that were not based on deep learning. (a) presents results over the MIT dataset, and (b) presents results over the ImgSal dataset. All algorithm saliency maps were unsmoothed.



(a) MIT



(b) ImgSal

Figure 2.8: spAUC scores for all deep learning-based algorithms tesed (with the cG curve for reference). (a) presents results over the MIT dataset, and (b) presents results over the ImgSal dataset. All algorithm saliency maps were unsmoothed.

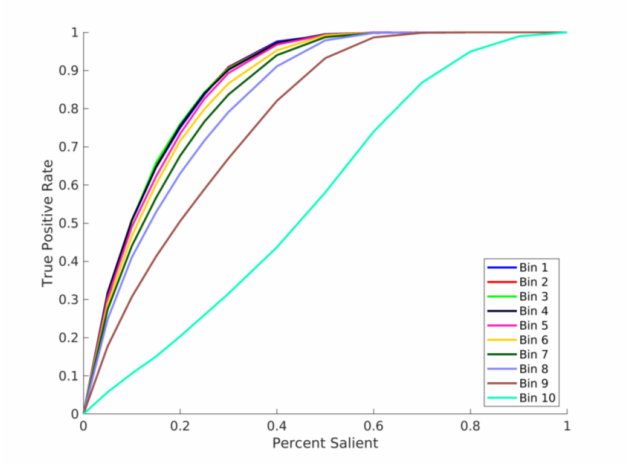
(Figure 2.7) and deep learning based models (Figure 2.8) for ease of comparison within these two groups.

As expected, the most extreme spatial bias is exhibited by the cG model (this is, after all, a prediction based solely on a spatial location), with an AUC very close to 1 for the central bins that then rapidly falls off to nearly zero in the more peripheral bins. Of the models not based on deep learning, GBVS exhibits the strongest degree of spatial bias. Also as expected, both $\mathrm{DGII_C}$ and $\mathrm{ICF_C}$ display a strong pattern of spatial bias, though it is interesting to see that the effect is much stronger for $\mathrm{ICF_C}$. This is could be a result of the fact that ICF is a weaker model that is more heavily reliant on a central bias to boost its performance, or it could be based on the fact that, as a model that utilizes only low-level features, it tends to produce maps that have a more diffuse estimate of saliency than the more object-centered maps produced by a model that makes use of higher-level features (like DGII). The saliency signal produced by ICF may therefore be more prone to being overridden by the central prior applied to $\mathrm{ICF_C}$.

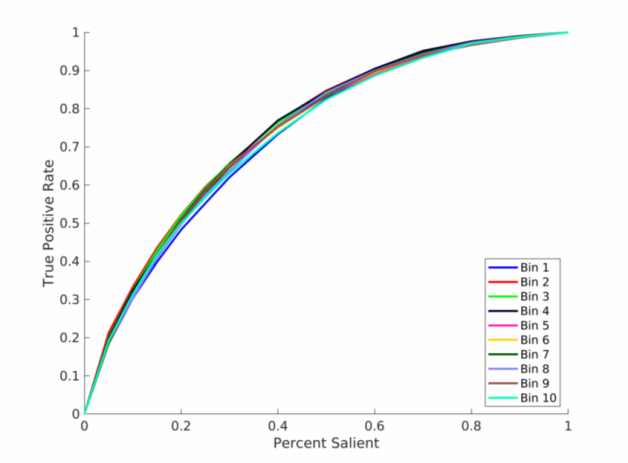
Surprisingly, although identified in [157] to have a potential peripheral bias in terms of raw saliency scores, in terms of predictive performance AWS is actually one of the least biased models. DGII, ICF, and SALICON all display strong spatial consistency on the MIT dataset, but interestingly all show a marked drop in the far periphery of the ImgSal dataset. It is possible this may be a result of the fact that the images in ImgSal are smaller than the images in the MIT dataset, and therefore the most peripheral bin may be prone to boundary effects for deep networks that have many processing layers. This is likely something that should be explored in more detail, however, because if it were entirely based on the boundary problem then one would expect ICF to show less impact than DGII and SALICON, but all three appear to show reductions of a similar magnitude.

Figure 2.9 shows the bin by bin ROC curves for GBVS (2.9a), AWS (2.9b), and a Gaussian centre prior (2.9c) for the MIT dataset. These figures show a more detailed view of the nature of the spatial bias in these various models, and these specific models were chosen for presentation in Figure 2.9 as they represent the most biased (GBVS) and most consistent (AWS) performance of the algorithms tested, as well as a representation of performance for a model that is only based on spatial location (cG).

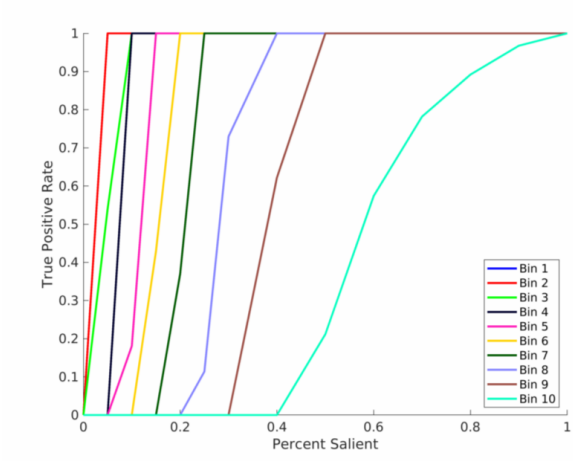
As mentioned in Section 2.3, one can calculate traditional AUC scores in a straightforward manner from the binned ROC results. We present the ordered ranking of unsmoothed algorithm



(a) GBVS



(b) AWS



(c) cG

Figure 2.9: spROC scores for GBVS, AWS, and a Gaussian centre prior on the MIT dataset. Of the models not based on deep learning, GBVS is the most spatially biased model tested, while AWS represents the most spatially consistent model tested.

MIT			ImgSal		
Model	AUC	σ	Model	AUC	σ
GBVS	0.82	0.093	GBVS	0.74	0.098
cG	0.81	0.188	CAS	0.71	0.057
CAS	0.78	0.049	cG	0.71	0.163
IKN	0.74	0.026	IKN	0.69	0.045
AWS	0.74	0.007	SSR	0.67	0.025
AIM	0.72	0.007	AWS	0.66	0.018
SSR	0.72	0.013	AIM	0.63	0.014
DVA	0.70	0.007	CVS	0.62	0.078
SUN	0.68	0.013	DVA	0.60	0.011
CVS	0.64	0.067	SUN	0.59	0.011
QSS	0.56	0.011	QSS	0.51	0.011
DGII_C	0.90	0.055	DGII_C	0.77	0.078
ICF_C	0.86	0.103	ICF_C	0.75	0.102
DGII	0.86	0.005	DGII	0.74	0.033
SALICON	0.83	0.009	SALICON	0.72	0.030
ICF	0.78	0.013	ICF	0.70	0.038

Table 2.2: Classical models are presented in the top of the table, and deep learning-based models at the bottom. Algorithms are ranked within their respective class by AUC score for the MIT and ImgSal datasets, presented along with the standard deviation calculated over bin scores representing the degree of inherent spatial bias. High performance on both data sets appears to be correlated with spatial bias.

performance over the MIT dataset in Table 2.2, along with the standard deviation of their binned AUC scores as a measure of the inherent spatial bias in each model. This provides a user with a direct performance measure (AUC score) that gives them a clear sense of algorithm performance operating over natural scenes, which is useful for any application in which choice of algorithm is solely dependent on its ability to predict human fixations in these environments. At the same time, we also have a quantifiable measure of how much of this performance is likely based on simple spatial bias versus an ability to identify salient visual stimuli, which is important for future scientific pursuits into saliency and saliency algorithm design.

We also present in Table 2.3 the AUC scores from the MIT dataset that have been weighted according to the relative image area occupied by each bin. The intention here is to provide a reasonable form of spatial correction, but which is transparent and deterministic in its source.

One such example of exploration into aspects of saliency algorithm performance is in the effect of smoothing kernel size. To explore this issue, we focused our efforts on the AIM algorithm as it has previously been shown to typically achieve maximum performance at relatively large

MIT		ImgSal	
Model	wAUC	Model	wAUC
AWS	0.73	SSR	0.65
CAS	0.71	CAS	0.64
SSR	0.71	AWS	0.63
IKN	0.71	IKN	0.63
AIM	0.71	AIM	0.62
DVA	0.69	GBVS	0.60
GBVS	0.69	DVA	0.59
SUN	0.66	SUN	0.59
cG	0.57	CVS	0.53
CVS	0.56	QSS	0.51
QSS	0.55	cG	0.50
DGII	0.86	DGII	0.69
DGII_C	0.82	SALICON	0.67
SALICON	0.82	DGII_C	0.65
ICF	0.77	ICF	0.65
ICF_C	0.72	ICF_C	0.61

Table 2.3: Classical models are presented in the top of the table, and deep learning-based models at the bottom. Algorithms ranked within their respective class by AUC score weighted by bin area for the MIT and ImgSal datasets. For models with low spatial bias (like AWS and AIM), there is little change in AUC score, while there is a significant drop in score for highly biased models (such as GBVS and CAS).

smoothing kernel sizes [1, 168]. However, as kernel size increases the numerical effects of border padding likewise increase, suggesting that at least some of these gains are due to the introduction of an implicit centre bias [221]. The exact degree to which improvements are due to the direct act of smoothing versus the introduction of spatial bias have previously not been quantified. Using spROC, however, we can directly explore this issue. Figure 2.10 displays the AUC scores by bin number for a range of different smoothing kernels acting on the AIM algorithm.

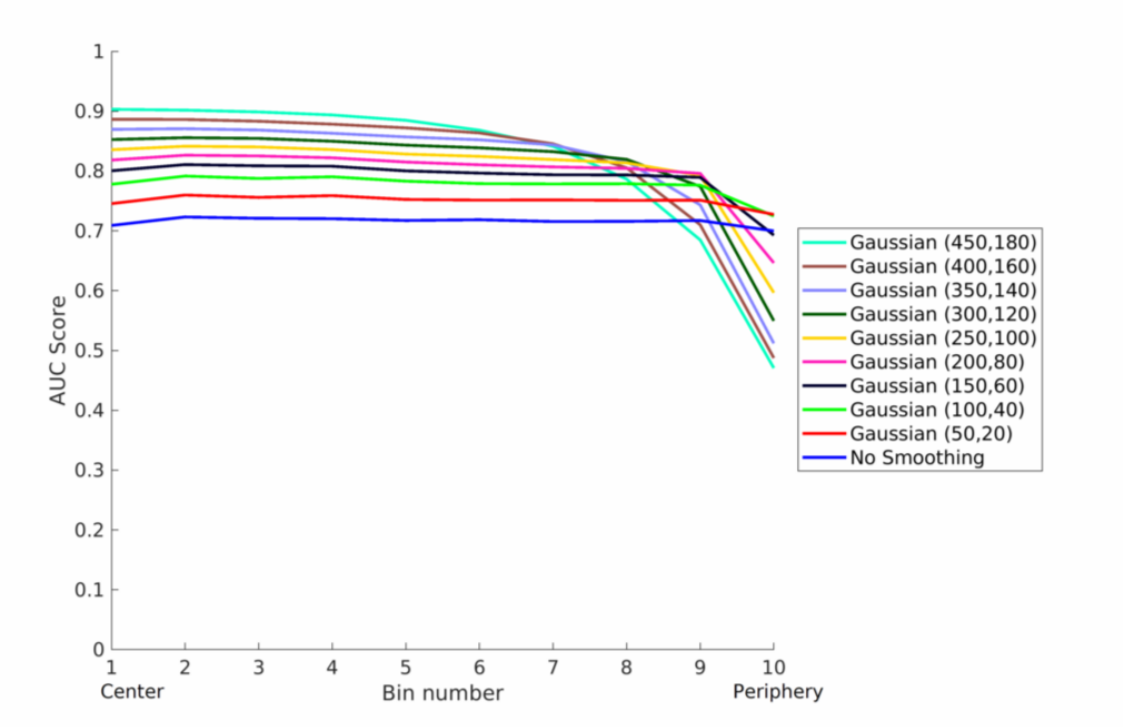


Figure 2.10: AUC score by bin for different degrees of smoothing applied to the AIM algorithm applied to the MIT dataset. Kernel properties are reported as $(size, \sigma)$. Initial smoothing boosts performance overall without appreciable increases in bias, but very large smoothing kernels sacrifice peripheral performance for central gains

At the smallest smoothing kernel tested, algorithm performance is almost uniformly boosted across all bins, including in the periphery. Subsequent smoothing initially boosts central scores without affecting peripheral performance, but a trade-off quickly develops thereafter between central gains and peripheral losses. Thus, we are able to begin to quantify the complex interactions smoothing has on the saliency signal, which opens the doors to further research into generally optimized post-processing techniques.

Although we have concentrated here on one particular form of spatial binning, it should be

straightforward to extend this methodology to explore other interesting aspects of saliency model performance. Of particular interest may be temporal binning, in which fixation points are binned by temporal order rather than spatial location.

2.5 Conclusion

Saliency algorithms are applied to a steadily increasing range of problems, and the pertinent aspects of performance will often change with the specific requirements of an application area. A primary difficulty in evaluating algorithm performance differences is the complicated interaction that visual appearance and spatial location have on salience. While it is true that traditional ROC metrics have a hard time fairly evaluating an algorithm's ability to identify visually distinct image elements given the sometimes overwhelming spatial component of the ground-truth set, discounting the role of spatial location in saliency can likewise lead to misleading conclusions regarding relative algorithm performance. This is particularly true for applications (such as image compression) in which gross predictive performance is more important than the underlying reason for why an element is salient.

We have presented here a novel evaluative method that provides insight into the impact of spatial location on algorithm performance. The method is flexible enough to be tailored for analyzing a wide range of aspects of algorithm performance, but can nevertheless be easily collapsed back into a straightforward measure of performance. We demonstrated a similar rank-ordering as found in the benchmark work of Judd *et al*. [1], but with added information specifying the spatial bias inherent to the tested algorithms. Further, we were able to directly explore the role of Gaussian smoothing on the spatial bias of an algorithm's performance. This provides us with the ability to begin quantifying *how* rather than simply *how much* smoothing modulates the saliency signal, which opens up a novel avenue of research into saliency algorithm optimization.

Chapter 3

SMILER: A Common Framework

The work described in this chapter has been published previously in the following format:

Calden Wloka, Toni Kunić, Iuliia Kotseruba, Ramin Fahimi, Nicholas Frosst, Neil D.B. Bruce, and John K. Tsotsos, "SMILER: Saliency Model Implementation Library for Experimental Research", *arXiv preprint*, 2018

The code for this project may be accessed in the following GitHub repository:
`https://github.com/tsotsoslab/smiler`
The contributions of the collaborating authors are noted below:

- *Toni Kunić*: Provided programming and software design support for the project, particularly in the implementation and extension of SMILER outside the MATLAB environment. Specific implementation contributions include:
  - 50% of software design for extending SMILER outside of the MATLAB environment
  - 65% of the coding for the dockerization of deep learning saliency models, Python architecture, and the CLI
- *Iuliia Kotseruba*: Provided testing and debugging support, as well as additional model dockerization. Specific implementation contributions include:
  - 15% of the coding for the dockerization of deep learning saliency models, Python architecture, and the CLI
- *Nick Frosst*: In partnership with Calden Wloka developed the early design plan for the MATLAB portion of SMILER, and provided testing and debugging support for the MATLAB portion of SMILER. Specific implementation contributions include:
  - 10% of the MATLAB implementation
- *Ramin Fahimi*: Provided testing and debugging support, as well as programming support for the extension from MATLAB to Python. Specific implementation contributions include:
  - 5% of the coding for the dockerization of deep learning saliency models, Python architecture, and the CLI
- *Neil Bruce*: Provided editorial and conceptual design feedback on the project.

Calden is responsible for directing and overseeing the development of SMILER, and for the development and articulation of its scientific goals and philosophy. In terms of implementation, he is also responsible for 90% *of the MATLAB implementation,* 50% *of the software design for extending SMILER outside of the MATLAB environment, and* 15% *of the coding for the dockerization of deep learning saliency models, Python architecture, and the CLI.*

The previous chapter introduced many of the issues inherent in evaluating saliency model performance with regards to predicting human fixation data. However, there are also a number of practical issues that also need to be addressed when attempting to compare and evaluate saliency models. In particular, just setting up and running a number of saliency models can pose a significant challenge. As discussed in Chapter 1, over the past two decades there has been a dramatic increase in both the number and nature of computational saliency models. Many of these models have been developed by distinct research labs and individuals, often with different degrees of programming experience and styles, and each model is typically developed for the express purpose of scientific research. Unfortunately, software developed primarily for research frequently lacks in-depth quality control, and is often difficult or time consuming to set up and execute, and may include undocumented assumptions, parameters, or conflicting requirements that present a major impediment to research sharing and reproducibility [222, 223]. Therefore, not only does the sheer volume of saliency models make it increasingly difficult for researchers to effectively explore and test the landscape of different approaches to saliency modelling, the lack of a standard interface to each model increases the likelihood that any given model may be incorrectly or erroneously configured, leading to mistaken or inconsistent results in the saliency literature.

For example, Table 3.1 shows scores computed by the similarity (SIM) metric [1] as calculated in three different studies (Vig *et al.* [3], Wang and Shen [191], and Berga and Otazu [224]) on the Toronto dataset [2]. Note that not only do the scores not match for even a single algorithm across any two studies, but also the rank order of performance shifts. For instance, Vig *et al.* find that their eDN model [3] outperforms the CAS model [143] both with and without added centre bias, whereas Wang and Shen find that CAS outperforms eDN. Similarly, Vig *et al.* find that with centre bias added, AWS [162] outperforms AIM [2] which outperforms GBVS [78], and without centre bias added the order shifts from best to worst to be GBVS, AIM, AWS. Neither ordering, however, agrees with the results of Berga and Otazu, who find a ranking of best to worst for these three models to be GBVS, AWS, and then AIM.

Note that this is not intended to accuse any authors of impropriety or misconduct, but rather to simply highlighting that without standardization three different studies give rise to three different sets of scores and rankings. Why this happens is best explained through an example: several saliency algorithms have been shown to operate best over colour spaces alternative to RGB, includ-

ing the Covariance-based Saliency (CVS) model [145], the Image Signature (IMSIG) model [168], and the Saliency Detection by Self-Resemblance (SSR) model [161] that all operate best in the CIELAB colour space, and the Quaternion-based Spectral Saliency (QSS) model [167] that performs best using YUV colour. However, the original model code released by each method's authors handles image input completely differently. CVS expects as input a string argument specifying an image path, then loads the image and converts it to CIELAB space internally. SSR expects as input an image variable in RGB format, which is converted to CIELAB space internally. IMSIG expects as input an image variable in RGB format, which is converted to CIELAB or DKL colour space when provided with an optional parameter setting (when no parameter is provided, IMSIG will process the image in RGB, despite the recommendation of the authors to use CIELAB space). QSS expects an image variable as input, but provides no internal image conversions; the model authors recommend that YUV format be used, and the conversion is expected to be performed by the user before calling the QSS model code. No one approach is any more correct than any other, but this lack of standardization places a non-negligible burden on users and can easily lead to errors or oversights in which a user believes they have configured the models to operate in the desired colour spaces but some subset of models is actually not being applied as expected. When one takes into account the number of additional parameters that must be controlled (such as numerical scaling of saliency map output, post-processing smoothing, the application of a centre prior, or any model-specific settings), the burden of use and chance for error is only further compounded. Likewise, for any hope of reproducibility, these parameters must all be exhaustively documented (which, unfortunately, has not always been the case within the literature).

Recent years have also seen a shift in model development toward methods that rely on deep learning networks. While many of these methods achieve very high benchmark performance, they also introduce a new practical challenge for the dissemination and sharing of code. In order to operate in a reasonable timeframe, most deep learning algorithms require a significant share of their computation to take place on a graphical processing unit (GPU). This necessitates that a user not only has access to GPU hardware, but also has the appropriate libraries installed that allow access to the GPU for calculations. Unfortunately, the setup processes of different GPU scientific computing libraries as part of the same development environment are often fairly involved and daunting for non-experts. Likewise, there is a lack of standardization, and frequently the

libraries necessary to run one model will be incompatible with the libraries required for another model. For example, oSALICON [220] is implemented in Caffe [225], while DeepGaze II [182] is implemented in TensorFlow [226]; as of the time of this writing, following the official setup documentation for one project interferes with the setup of the other. While running both libraries on the same system is possible, it requires knowledge that goes beyond the official documentation. Therefore, not only will the installation of even a single model likely be a large barrier of entry for a user who is not actively pursuing work with deep learning-based development, but also providing simultaneous access to a general-purpose library of saliency models is extremely difficult without isolating incompatible model dependencies from each other. Due to potentially frail assumptions regarding backward compatibility, there is a significant risk that important contributions may be lost to time or not explored in sufficient detail owing to the need to operate within a specific ecosystem.

This work aims to facilitate research efforts in computational salience by addressing these software challenges. We do so by introducing the Saliency Model Implementation Library for Experimental Research (SMILER). SMILER provides library-like functionality for saliency models, standardizing the input, output, and parameter specifications for each model, and isolating incompatible model components from each other. At the time of this publication SMILER supports twenty three models: Attention by Information Maximization (AIM) [2], Adaptive Whitening Saliency (AWS) [162], Boolean Map Saliency (BMS) [152], Context Aware Saliency (CAS) [143] based an open implementation [219], Covariance-based Saliency (CVS) [145], DeepGaze II (DGII) [182], Deep Visual Attention Prediction (DVAP) [191], Dynamic Visual Attention (DVA) [160], Ensemble of Deep Networks (eDN) [3], Fast and Efficient Saliency Detection (FES) [227], Graph-based Visual Saliency (GBVS) [78], Intensity Contrast Features (ICF) [184], the Itti-Koch-Niebur Saliency Model (IKN) [25], Image Signature (IMSIG) [168], Learning Discriminative Subspaces (LDS) [165], a Deep Multi-Level Network (MLNet) [188], an open implementation [220] of Saliency in Context [7] (oSALICON), Quaternion-Based Spectral Saliency (QSS) [167], RARE2012 [148], Saliency Attentive Model (SAM) [186], Saliency Detection by Self-Resemblance (SSR) [161], Saliency using Generative Adversarial Networks (SalGAN) [189], and Saliency Using Natural statistics (SUN) [159]. This set provides a broad representative sample of models popular in the saliency research community focused on fixation prediction, and the system is designed to be easily extensible with additional

Reference	[3]	[191]	[224]
eDN [3]	0.573/0.487	0.40	-
CAS [143]	0.555/0.427	0.44	-
AWS [162]	0.558/0.407	-	0.352
GBVS [78]	0.534/0.496	0.49	0.397
AIM [2]	0.549/0.426	0.36	0.314
IKN [25]	-	0.45	0.366

Table 3.1: An example of inconsistent model results. Here we show the SIM [1] scores for five algorithms over the Toronto dataset [2] as reported by three recent publications (note that [3] report two scores, one with added centre bias and one without). - indicates that a model was not run in that particular study. Note that we are not claiming any wrong-doing on the parts of these studies, but rather pointing out that each study likely executed these models in slightly different ways, leading to inconsistent results and a substantial challenge for reproducibility in the literature.

models.

It should be noted that although the current collection of models supported by SMILER focuses models that perform pixel-wise assignment of conspicuity values which have been predominantly applied to the domain of human fixation prediction, there are, as noted previously in this dissertation, other branches of saliency model research, such as salient object detection (*e.g.* see [93] for an early example, and [228] for an overview and recent survey). Likewise, the models included are predominantly focused on saliency prediction over static scenes, but there is nevertheless significant interest in saliency over dynamic stimuli (*e.g.* see [120, 121, 122]). This focus on models that are more representative of fixation prediction over static images is not intended to dismiss or ignore these other research avenues, but rather is meant to form a solid base for the SMILER platform.

3.1 Existing Repositories and Bundles

The usefulness of a common resource for providing some central structure to saliency research has been previously recognized, and a number of efforts have been implemented in this vein.

3.1.1 The MIT Saliency Benchmark

Probably the most extensive and influential of the centralized saliency research projects, the MIT Saliency Benchmark is the work of Bylinski *et al.* [79]. The primary focus of the Benchmark is on evaluation and the comparative ranking of model performances in free viewing prediction. Given

this concentration on performance evaluation, the Benchmark does not directly host model code but rather links, when possible, to the individually hosted code repositories of each author. The Benchmark does host two unique testing datasets: the MIT300 test set [1] and the CAT2000 test set [12]. Note that these sets are distinct from the MIT1003 and the CAT2000 training sets used in other parts of this work; each test set is provided by the same researchers as the respective MIT1003 or CAT2000 training set using the same data gathering conditions, but the ground-truth data is not publicly available. This withholding of ground-truth fixations is intended to prevent cheating by training over the test sets directly. Saliency researchers are requested to submit their saliency maps for each test image set, and the maintainers of the Benchmark will score them according to a set of commonly used metrics.

The MIT Saliency Benchmark provides a useful first stop when searching for available saliency models (due to the numerous links to code and papers), as well as a systematic ranking of algorithm performance for predicting fixation locations in natural images. It serves as a curated repository for the evaluation of free-viewing fixation prediction, as well as a number of potentially useful links for the topic of saliency research. Given its focus on a non-public benchmarking set, however, the Benchmark is less able to support exploratory research with saliency models. Researchers looking to apply models to different datasets and tasks must accept nearly all of the onus of determining how to run and apply code with a wide range of implementation styles and formats. The MIT Saliency Benchmark, therefore, can be viewed as a complementary effort to SMILER.

3.1.2 The Saliency Toolbox

The Saliency Toolbox is a MATLAB-based code base containing both the GBVS and IKN saliency models, as well as some visualization tools, proto-object detection code, and rudimentary graphical interfaces [90]. It has been widely used as an accessible set of tools for computing saliency maps. Although it is relatively well documented and straightforward to set up and use, the primary limitation of the Saliency Toolbox is its specificity to only a small number of saliency algorithms. SMILER attempts to retain the accessible implementation of the Saliency Toolbox while providing a much more comprehensive set of available models and a clear pattern for extension to future developments.

3.2 SMILER's Contributions

SMILER provides two primary benefits to the saliency research community: reducing the burden of use for code execution, and promoting the consistency and reproducibility of experiment results. The first step in achieving these goals is the establishment of a common model-independent application programming interface (API). In order to make this API as effective as possible in facilitating a wide range of research, we ensure the following qualities:

- It should be possible to run each algorithm in a default mode that requires minimal user input or selection of settings, providing an intuitive mode that can be used without expert-level algorithmic familiarity. At the most basic level of function, a model should expect only that an input image is specified, and it should return as output a saliency map corresponding to that image. By default, this saliency map should be the same height and width as the input image.

- As much as possible, there should be no loss in the flexibility of parameter options available for each individual model. While it is not possible to have a common set of parameters for each algorithm, to reduce the complexity of operation as much as possible it should be possible to selectively choose which parameters to manually specify, with unspecified parameters automatically populated with default values (thereby allowing for a smooth transition from fully default mode through to fully user-specified operations).

By creating a standard interface for model execution, we allow users to learn a single API rather than one for each model. The flexible method for parameter specification allows researchers to engage with models at a variety of levels of depth, from novel benchmarking work using default settings through to the analysis of model behaviour over a range of parameter settings.

By standardizing model execution, we also ensure that when researchers run a given model with particular settings, they are sure to get the same results as when another researcher runs the same model with the same settings. If both researchers were expected to independently set up and execute the model using their own custom scripts, it is entirely possible for unintentional bugs or oversights to lead to inconsistencies between them. Of course, it is entirely possible for SMILER to contain bugs, but by fostering an open and centralized repository for saliency model code, we

ensure that when bugs are found and corrected this correction is distributed to all users.

With a straightforward and flexible code base for easily executing a large ecosystem of saliency models, we envision a number of research directions that SMILER can support, including but not limited to:

- Performance benchmarking on applications outside of fixation prediction for which saliency may be applicable, but extensive performance testing is not currently available. Examples include:
  - Anisotropic image or video compression (*e.g.* [132, 133, 27, 135])
  - Defect detection (*e.g.* [229, 230])
  - Image cropping (*e.g.* [231]) or retargeting (*e.g.* [232])
  - Image domains outside the natural images which form the bulk of fixation datasets, such as websites [233] or satellite imagery [234]
  - Image quality assessment (*e.g.* [129, 130, 131])
  - Robotic navigation (*e.g.* [28, 29]) or search (*e.g.* [30])
- Saliency model evaluation for other attentional aspects beyond fixation prediction, such as the psychophysical evaluations proposed in Bruce *et al.* [157].
- Increasing the robustness of conclusions for research which compares experimental findings in psychology or neuroscience to saliency algorithms (*e.g.* [52, 112, 235]) by allowing comparison against many saliency models rather than a single one.

3.3 Design Overview

In order to leverage as wide a range of existing saliency model implementations as possible, as well as to support researchers with different degrees of computational and software resources available to them, SMILER is comprised of two major programming language components: a MATLAB component and a command-line interface (CLI) implemented in Python. The MATLAB component comes with a subset of available models and is fully cross-platform so long as the computer supports the MATLAB environment and the user has access to the appropriate licenses for MATLAB and any specific toolboxes required by a given model. The CLI is currently only supported for the Linux

operating system, but provides access to the full suite of SMILER models, both MATLAB and deep learning. In order to foster open software development and move away from proprietary software systems, the CLI will be the primary focus of future development for the SMILER project, with an emphasis on adding models that do not depend on a MATLAB license. To minimize code drift across multiple interfaces, all models included in SMILER contain a configuration and information file described in Section 3.3.1.

Prior to the shift in algorithm development toward deep learning models, the majority of saliency models were released for the MATLAB programming environment. As a consequence, early development of SMILER was also based in MATLAB. However, the need to handle deep learning models that are predominantly implemented in languages other than MATLAB necessitated a shift to another language. Nevertheless, it was felt that it would not be desirable to drop the MATLAB specific structure that is already in place, as there are many users who would prefer to operate within the MATLAB environment (for example, many researchers are already familiar with MATLAB through the use of tools such as the PsychToolbox [236], and may prefer to keep their research efforts in the same programming environment). Therefore, the design of SMILER retains MATLAB functionality for all algorithms available in the MATLAB environment, as well as a functional MATLAB interface for executing these models. The SMILER CLI utilizes the MATLAB's Python API to allow invocation of MATLAB models in the background, without using the full MATLAB graphical user interface.

Whether one is working through MATLAB or the SMILER CLI, the general principles of SMILER operation remain the same, and the details of operation are kept as close as possible given the different nature of the MATLAB Integrated Development Environment (IDE) and the CLI. An overview of operation for the MATLAB interface is given in Section 3.3.2, and for the SMILER CLI in Section 3.3.3. Due to the more extensive support of saliency models and support for YAML-based experiment specification (discussed more thoroughly in Section 3.3.3), we would encourage users to preferentially use the CLI.

SMILER attempts as closely as possible to maintain the originally intended functionality of each model. However, there are times when this is not possible. For example, although expecting the output map to be the same height and width as the input image seems like a straightforward assumption, it is not the default behaviour of all algorithms. Some models automatically resize

input images to a specified size, and return this size as output, whereas others such as SUN [159] make a point of returning only the portion of the image for which output is valid without image padding (trimming the half-width of the feature kernels from the image border). Although this inconsistency between input and output size may be a distinct choice by the model designers with a clear justification, for the purposes of SMILER it was felt that ensuring a common behaviour across algorithms was the more important consideration and therefore SMILER will re-scale or pad as appropriate the saliency maps to be the same dimensions as the original input image.

Models for which the full source code has been released are preferred candidates for inclusion in SMILER, as this allows for more robust crowd-sourced bug checking, access to the full range of algorithm parameters (particularly for post-processing steps such as smoothing), and aids in future code maintenance (for example, the use of deprecated functions that are no longer supported by MATLAB or third party libraries). It should be noted, however, that several models are nevertheless included despite only having access to a pre-compiled version, namely AWS [162], FES [227], and RARE2012 [148]. The pre-compiled version of CAS [143] provided by the original study authors is not compatible with SMILER, and therefore an open source implementation [219] has been used. In a similar vein, code for the SALICON model [7] is not available at the time of this writing, but we include the oSALICON [220] implementation that is based on the original model.

3.3.1 A Common Format for Information and Configuration

There are a number of parameters and controls for pre- and post-processing of saliency maps that are common to many or all models. As well, each model in SMILER requires several important attributes to be associated with it, including citation information and model-specific parameters. In order to provide this information in a manner that is extensible to the inclusion of future model properties or specifications and independent of the specific programming interface accessing the model, several JavaScript Object Notation (JSON) configuration files are used. JSON is an open standard for providing human-readable attribute-value pairs, and provides an effective format for storing model information in SMILER.

Parameters that affect the execution of a majority of SMILER models are referred to as *global* and are described in a `config.json` file in the root of the SMILER directory. These parameters and their default values are shown in Table 3.2. Listing 3.1 shows an example JSON parameter

specification. The user shouldn't need to modify these JSON files directly, as they contain specifications for the SMILER system. The user should specify parameters at run-time via the SMILER CLI or MATLAB interface, or via YAML experiment files.

Parameter	Default	Valid Values	Description
color_space	"default"	"RGB", "gray", "YCbCr", "LAB", "HSV"	Specification for pre-processing conversion of the image colour channels.
center_prior	"default"	"default", "none", "proportional_add", "proportional_mult"	Specification for the addition of a priori bias toward the centre.
center_prior_prop	0.2	Float between 0 and 1.	Specifies the sigma of the Gaussian as a proportion of image dimensions. Only used when center_prior is set to proportional_add or proportional_mult.
center_prior_weight	0.5	Float greater than 0.	Specifies the weight given to the prior in the form of multiplicative gain. Only used when center_prior is set to proportional_add or proportional_mult.
center_prior_scale_first	true	Boolean.	If true, first scale the saliency map to be between 0 and 1 before adding the prior.
do_smoothing	"default"	"default", "none", "custom", "proportional"	Specification for post-processing smoothing.
smooth_size	9	Integer greater than 0.	Custom smoothing kernel size, only used when do_smoothing is set to custom.
smooth_std	3.0	Float greater than 0.	Custom smoothing kernel standard deviation, only used when do_smoothing is set to custom.
smooth_prop	0.05	Float greater than 0.	Proportional smoothing kernel parameter, only used when do_smoothing is set to proportional.
scale_output	"min-max"	"min-max", "none", "normalized"	Specification for rescaling saliency map values into a specified range.
scale_min	0.0	Float less than scale_max.	Minimum saliency value in the map, only used when scale_output is set to min-max.
scale_max	1.0	Float greater than scale_min.	Maximum saliency value in the map, only used when scale_output is set to min-max

Table 3.2: Default values for global SMILER parameters, as defined in `config.json`.

Listing 3.1: The global parameter dictionary contained in `config.json`.

```
"do_smoothing": {
  "default": "default",
  "description": "Specification for post-processing smoothing.
    'default' uses whatever smoothing step is provided by the originally
     released model code.
    'none' turns off post-processing smoothing (though it should be noted
     that some implicit smoothing (eg. through image resizing) will remain
     ).
    'custom' smooths the map with a specified kernel.
    'proportional' smooths the map with a kernel sized to the major
     dimension of the image.",
  "valid_values": ["default", "none", "custom", "proportional"]
}
```

As can be seen, `parameters` are defined as a nested dictionary. Each parameter is populated with three fields: `default`, `description`, and `valid_values`. The `default` field is used when no other source of parameter specification is available. The `description` and `valid_values` fields are intended for human consumption; each SMILER interface provides a method for accessing and displaying this information to a user (detailed in Section 3.3.2 for the MATLAB interface and Section 3.3.3 for the CLI). The `description` field provides a brief explanation for the role the parameter plays in the calculation of a saliency map, while the `valid_values` field provides either an explicit set of available parameter assignments (*e.g.* for the `scale_output` parameter there are three options: `min-max`, `none`, or `normalized`) or a specified range (*e.g.* an "Integer greater than 0" for `smooth_size`).

Listing 3.2: An example `smiler.json` file showing model-specific information for the AIM algorithm. [2]

```
{
  "name": "AIM",
  "long_name": "Attention by Information Maximization",
  "version": "1.0.0",
  "citation": "N.D.B. Bruce and J.K. Tsotsos (2006). Saliency Based on
   Information Maximization. Proc. Neural Information Processing Systems (
   NIPS)",
  "model_type": "matlab",
  "model_files": [],
  "parameters": {
    "AIM_filters": {
      "default": "21jade950.mat",
      "description": "The feature filter set to be used by the AIM
       algorithm. In the form [size][name][info], where each filter is
       size by size in dimension, name is the ICA algorithm used to derive
        the filters, and info provides a measure of the retained
```

```
            information (higher numbers correspond to more filters).",
12          "valid_values": [
13            "21infomax[900,950,975,990,995,999].mat",
14            "21jade950.mat",
15            "31infomax[950,975,990].mat",
16            "31jade[900,950].mat"
17          ]
18      }
19    }
20  }
```

Each model includes additional model-specific information in a `smiler.json` file included in the root of its subfolder. An example `smiler.json` file is shown in Listing 3.2.

As can be seen, each file contains information providing both the SMILER shortened designation for the model (in this case, `AIM`) as well as its full name and citation information. `model_type` allows the code to easily check whether pre-requisites are available to execute the code (for example, if the MATLAB engine is not installed, SMILER will skip MATLAB-based models with a warning rather than an execution error). `model_files` provides SMILER with a list of any files required for the model execution (*e.g.* network weights for a CNN-based model). Model-specific parameters are specified using the same system as the global parameters in `config.json`. Additionally, some models contain a `notes` field that includes human-readable information pertinent to the specific model (such as recommendations by the original model authors or additional information that may be of use to a user).

SMILER is programmed to take a flexible approach to parameter specification, populating parameter fields according to a priority order. This order is, from greatest to least precedence: user specified values (provided at runtime, or via YAML experiment file), model-specific default values (defined in model's `smiler.json` specification), and global default values (defined in SMILER's internal config.json).

3.3.2 Overview of MATLAB Interface

In order to help users navigate and use the code base provided by SMILER, a number of helper functions are provided. This section describes the supporting code base for the MATLAB portion of SMILER; the suite of tools that support the CLI are described in Section 3.3.3.

The primary helper file is the installation file, `iSMILER.m`. This file adds all other helper func-

tions and all bundled MATLAB-based models to the MATLAB `path`. By default, this installation will not save the changes to the `path` beyond the current session, but a user may optionally specify that `path` changes should be permanent by calling:

```
1    iSMILER(true);
```

Should users have permanently modified the `path` and later change their mind, SMILER `path` changes may be undone by using the uninstall function provided in the file `unSMILER.m`.

The function `smiler_info` provides a text interface in MATLAB for a user to query parameter information. This may be called without any arguments or using the string argument `'global'` to receive information about global parameters, or a specific MATLAB-based model may be specified as the input argument and the model-specific parameter and citation information for that model will be displayed.

In order to bring each included algorithm into compliance with the common API of SMILER, the code for each model is encapsulated in a wrapper function with the format *`[model_name]`*`_wrap.m`, where *`[model_name]`* is a string selected from the following available list of included algorithms:

- `AIM`: Attention by Information Maximization [2]
- `AWS`: Adaptive Whitening Saliency [162]
- `CAS`: Context Aware Saliency [143], using the implementation by [219]
- `cG`: A centered Gaussian prior
- `CVS`: Covariance-based Saliency [145]
- `DVA`: Dynamic Visual Attention [160]
- `FES`: Fast and Efficient Saliency [227]
- `GBVS`: Graph-Based Visual Saliency [78]
- `IKN`: The Itti-Koch-Niebur Saliency Model [25]
- `IMSIG`: Image Signature [168]
- `LDS`: Learning Discriminitive Subspaces [165]
- `QSS`: Quaternion-Based Spectral Saliency [167]
- `RARE2012`: A multi-scale rarity-based saliency model [148]
- `SSR`: Saliency Detection by Self-Resemblance [161]
- `SUN`: Saliency Using Natural statistics [159]

Each function operates with the following function call:

```
output_map = [MODEL_NAME]_wrap(input_image,
                               params)
```

where the `input_image` is either a string specifying the file path of an image or is a variable containing image data, and `output_map` is a single-channel saliency map with the same height and width as the image specified by `input_image`. `params` is an optional input variable in the MATLAB structure format that provides a mechanism for specifying parameter values. As mentioned in Section 3.3.1, every model's behaviour is governed by a set of parameters that are specified as key-value pairs. If no parameter structure is provided, the wrap function will automatically populate the parameter settings with default values appropriate for the given model. If some but not all parameters are specified in the input, then the wrap function will likewise operate with default values for any unspecified structure elements.

A basic example showing the explicit calculation of four models (AIM, AWS, IKN, QSS) on an input image specified by a path string is given in Listing 3.3.

Listing 3.3: Example MATLAB script showing the calculation of saliency maps for the AIM, AWS, IKN, and QSS models.

```
img_path = 'path/to/example.png';

AIM_map = AIM_wrap(img_path);
AWS_map = AWS_wrap(img_path);
IKN_map = IKN_wrap(img_path);
QSS_map = QSS_wrap(img_path);
```

This can be written more conveniently as a loop, iterating over the same set of models and executing each in turn and saving the saliency map as a separate image.

Listing 3.4: Example MATLAB script using SMILER to calculate saliency maps for the AIM, AWS, IKN, and QSS models.

```
models = {'AIM', 'AWS', 'IKN', 'QSS'};

img_path = 'path/to/example.png';

for i = 1:length(models)
  salmap = feval([models{i}, '_wrap'], img_path);
  imwrite(salmap, [models{i}, '_saliency_map.png']);
end
```

Note that the code makes use of the MATLAB `feval` function to dynamically execute code based on a string argument, which allows for a simple interface for scripting and batch execution.

Sample 3.4 can be easily extended if specific parameters for some models are desired. For example, if a user wanted to use one of the other learned ICA filter bases for the AIM algorithm and wanted QSS to operate over the HSV colour space (but were otherwise fine with all other default parameters), then the modified version of the script shown in Listing 3.5 could be used.

Listing 3.5: Example MATLAB script using SMILER to calculate saliency maps for the AIM and QSS models with customized parameters.

```
models = {'AIM', 'AWS', 'IKN', 'QSS'};

img_path = 'path/to/example.png';

for i = 1:length(models)
  params = struct();
  switch(models{i})
    case 'AIM'
      params.AIM_filters = '21infomax999.mat';
    case 'QSS'
      params.color_space = 'hsv';
  end
  salmap = feval([models{i}, '_wrap'], img_path, params);
  imwrite(salmap, [models{i}, '_saliency_map.png']);
end
```

Note that whether the parameter is model-specific or global, the method of user specification is the same (in this example the user specifies AIM's model-specific parameter `AIM_filters`, whereas for QSS it is the global parameter `color_space` that is specified).

All the above samples assumes that the `iSMILER` function has already been run, and therefore all wrapper functions are available on the MATLAB `path`. Additional examples are available as part of the SMILER GitHub repository.

3.3.3 Overview of SMILER CLI

Although the MATLAB interface is fully functional and supports all MATLAB-based models, the CLI is the recommended method of use, and future extensions to the SMILER library will likely be focused in this direction. Not only does this help migrate SMILER away from software requiring a proprietary license (MATLAB), but it also provides a more flexible platform for extension and

experiment design that better supports protocol documentation.

The SMILER CLI is based on a core structure of functions that provide an interactive text-based interface to users. This includes commands to manage the containerized images for the available non-MATLAB models (see Section 3.4.2 for more details on model isolation and containerization), which at the time of this writing include the following models:

- `BMS`: Boolean Map Saliency [152]
- `DVAP`: Deep Visual Attention Prediction [191]
- `DGII`: DeepGaze II [182]
- `eDN`: Ensemble of Deep Networks [3]
- `ICF`: Intensity Contrast Features [184]
- `MLNet`: Deep Multi-Level Network [188]
- `oSALICON`: Open-source Saliency in Context [220], based on the original model by [7]
- `SAM`: Saliency Attentive Model [186]
- `SalGAN`: Saliency using Generative Adversarial Networks [189]

SMILER's CLI is designed to function in a Linux environment. The library is interacted with using commands with the following pattern:

```
1    smiler COMMAND [OPTIONS] [ARGS]
```

where `[OPTIONS]` and `[ARGS]` are command-specific options and arguments to modify program behaviour. The SMILER commands available are as follows:

- `clean`: Deletes downloaded files and docker images.
- `download`: Downloads model files and docker images.
- `info`: Provides information on SMILER models.
- `run`: Runs model(s) on images in a directory.
- `shell`: Runs a shell interface appropriate to the model environment.
- `version`: Displays SMILER version information.

Further information about the usage of any command can be obtained by appending the `--help` flag.

Although users may directly use the CLI to conduct experiments and generate saliency maps with SMILER, the CLI additionally supports experiment specification using YAML. This is the

recommended method of operation, as it allows a user to maintain explicit records of experimental settings and protocols through stored YAML specification files.

YAML is a data serialization language designed to be easily written, read, and understood by humans. SMILER uses YAML files to specify experiments. These YAML specification files are composed of two sections: an `experiment`, which provides global specification details, and one or more experimental `runs`, which provide details for a specific algorithm call. An example is presented in Listing 3.6.

Listing 3.6: An example YAML specification file

```
1  experiment:
2    name: Example 1
3    description: An illustrative example of how to set up SMILER YAML
       experiments.
4    input_path: /tmp/test_in
5    base_output_path: /tmp/test_out
6    parameters:
7      do_smoothing: none
8
9  runs:
10   - algorithm: AIM
11     output_path: /tmp/AIM_smoothing
12     parameters:
13       do_smoothing: default
14
15   - algorithm: AIM
16     output_path: /tmp/AIM_no_smoothing
17
18   - algorithm: DGII
19
20   - algorithm: oSALICON
21     parameters:
22       color_space: LAB
```

The `name` and `description` fields are primarily for user records, and facilitate organization and sharing of experimental protocols by providing a lightweight document that can easily be created and stored for each experiment conducted and run on any system with SMILER installed. `input_path` is the folder that contains the images to be processed in this particular experiment. `base_output_path` provides a root location for output maps to be saved, which by default will be placed in a subfolder at this location named for the algorithm that produced it (*e.g.* in listing 3.6 DGII and oSALICON will be saved in `/tmp/test_out/DGII` and `/tmp/test_out/oSALICON` respectively).

YAML specification introduces an additional layer to parameter precedence. The `parameters` field within the `experiment` field provides a way to set customized values that will be used for all runs, but these may be overridden for a specific run by adding a `parameters` field to that run. This is demonstrated in the example shown in Listing 3.6; all runs are set to be performed without smoothing based on the parameter specification under the `experiment` field, but the first run using AIM overrides this specification and instead uses default smoothing parameters. In this case, both AIM runs include `output_path` fields that will override the default behaviour using `base_output_path`. In the provided example, DGII will be run without any additional specifications beyond those provided in the `experiment` field, while oSALICON will be run with an additional specification of the `color_space` parameter (since there is no `color_space` specification under `experiment`, all other runs will use the built-in SMILER default: RGB).

3.4 Further Implementation Details and Requirements

3.4.1 MATLAB Implementation

In addition to the interface functions mentioned in Section 3.3.2, a number of support functions are provided that are used internally by either the function wrappers or user-level helper functions (such as image loading functions that will work with either a path specification or an array variable containing image data). These functions are primarily intended for SMILER's internal use, and therefore users should not expect to interact with them directly. This includes the `jsonlab` toolbox [237] for interacting with the configuration files.

We recommend using `2016a-2017b` versions of MATLAB. Other versions may not fully support all available MATLAB models or SMILER code (for example, the pre-compiled AWS model, for which no source code is available, does not function in newer versions of MATLAB due to the deprecation of the `princomp` function).

3.4.2 Command Line Interface (CLI)

As previously mentioned, deep learning libraries are not always compatible on the same system, which presents a challenge for executing deep learning-based saliency models that rely on incompatible libraries. To solve this issue, SMILER makes use of *containerization*, which is also known as

operating system level virtualization. This method creates isolated user-space instances called *containers* that share the same OS kernel and drivers, but are otherwise separated. Thus, each model may be fully encapsulated within its own container, isolating any system-level libraries that may interfere with those used by other model implementations. In addition to granting this isolation, the container may be designed to provide a full specification of a model's software requirements that will be downloaded and installed upon container instantiation without the necessity of user input. This alleviates the (sometimes significant) challenge of installing all required dependencies for a given model, and allows any model encapsulated in this way to be called using a common format, analogous to the functional wrapping described in Section 3.3.2.

SMILER accomplishes containerization using Docker [238], and its extension `nvidia-docker` [239] which supports GPU computing. The only other dependencies for using the SMILER CLI are Python and the `click` and `yaml` Python modules.

Note that a GPU is required to efficiently run the deep learning-based models supported by SMILER. It may be possible to run all models with a compatible graphics card with at least 4GB of memory, though it is highly recommended that one use a system with 6GB or more of GPU memory.

It should be mentioned that, although all efforts have been made to eliminate code duplication in order to avoid implementation drift between the MATLAB and Python code bases, there are some support functions in the SMILER MATLAB suite that had to be re-implemented in Python in order for the CLI portion of SMILER to be able to operate independently of a MATLAB license. To ensure that these processing steps are equivalent, we maintain a set of unit tests that can be used to ensure these processing steps remain equivalent in face of future improvements to the SMILER code or new MATLAB versions.

3.5 Extending SMILER

SMILER is designed to be easily extended with new models. The manner in which new models may be added to the library depends on whether the model runs through MATLAB or not.

3.5.1 Adding MATLAB Models

MATLAB models are added to SMILER by creating a subfolder within the SMILER directory with the following path structure: `[SMILER_root]/models/matlab/[MODEL_NAME]`, where `[SMILER_root]` indicates the path to the primary root folder for SMILER and `[MODEL_NAME]` is the designation by which SMILER will refer to the new model (for example, AIM is the SMILER designation for the Attention by Information Maximization model [2]). This directory will hereafter be referred to as the *model directory*.

When adding a new model to SMILER, all files necessary to execute the model should be placed in the model directory or subfolders of this directory[1], while extraneous files (such as example images or example execution code) should be removed to avoid cluttering the repository. In addition to the original source code for the model, two new SMILER-specific files must be created: the wrapper function (`[MODEL_NAME]_wrap.m`) and the `smiler.json` information file. Writing a wrapper function will be discussed first, followed by details for populating a `smiler.json` file.

A wrapper function skeleton is provided as a template, and can be found in the SMILER code base in
`[SMILER_root]/examples/model_skeletons/MATLAB/skeleton_wrap.m`. The contents of this file are displayed in Listing 3.7.

The comment block at the top of the `skeleton_wrap.m` file should be updated, replacing all instances of `[skeleton]` and `[SMILER name]` with the SMILER designation for a given model. Additional fields which should be replaced for better record keeping include `[full model name]` with the full name of the model, `citation information for the model` with full bibliographic information for the model, and `[wrap author]` with the name of the person creating the wrap function. Contact information may also optionally be added beneath the author name.

Listing 3.7: Contents of `skeleton_wrap.m`, a template for MATLAB model wrapper functions.

```
1 % [skeleton]_wrap executes the [full model name] ([SMILER name])
2 % saliency model in the SMILER format. If you use results produced
3 % by this model, please cite the following paper:
4 % [citation information for the model]
5 %
6 % Wrap code written by: [wrap author]
```

[1] There may occasionally be exceptions, such as in the case of the GBVS and IKN models which both rely on a shared code base. In this case, the GBVS model directory contains all model files, and the IKN wrapper function has been specifically set up to point to these files.

```
7  %
8  % * Function Syntax:
9  % salmap = [SMILER name]_wrap(input_image, params)
10 % **** Input ****
11 % * input_image = Either the file name of an image to analyze or the image
12 % matrix
13 % * params = A structure variable that allows the user to control any
14 % algorithm-specific tunable parameters. The algorithm will default to
15 % using a set of default parameters defined by the model's authors. To see
16 % a list of available model-specific parameters, use the function call:
17 % smiler_info('[SMILER name]')
18 % **** Output ****
19 % * salmap = A matrix representing saliency map values across the visual
20 % field produced by the AIM model
21 function salmap = [SMILER name]_wrap(input_image,in_params)
22
23 %% Setting up parameters
24 if(nargin < 2)
25     in_params = struct();
26 end
27
28 % check parameter fields which are the same across all models, assign
    defaults to missing fields
29 params = checkCommonParams(in_params);
30
31 % get the path to smiler.json
32 pathroot = mfilename('fullpath');
33 [pathroot, ~, ~] = fileparts(pathroot);
34
35 % check the model-specific parameter fields
36 params = checkModelParams(params, [pathroot, '/smiler.json']);
37
38 % if the model has any parameters which conflict with how SMILER handles
39 % parameters, these should be dealt with here. See GBVS_wrap for an
40 % example.
41
42 %% Reading the image
43 % if the model expects a three-channel image, it must enforce that by
44 % passing 'true' as a third argument into the checkImgInput function
45 % if the model expects a non-RGB colour space by default, this value
    should
46 % be assigned to params.color_space before invoking the checkImgInput
47 % function
48 img = checkImgInput(input_image, params.color_space);
49
50 %% Calculating the saliency map
51
52 % make a call to the saliency calculation function, as well as check for
53 % any default pre- or post-processing which is not included in the
    saliency
54 % calculation function
55
56 % perform any post-processing which is being handled by SMILER
57 salmap = fmtOutput(salmap, params);
```

Much of the required execution code is already provided in the `skeleton_wrap.m` file. This code roughly breaks down into three major steps: setting up parameters (Lines 23-41), reading the image (Lines 42-49), and calculating the saliency map (Lines 50-58).

Setting up parameters (Lines 23-41) first checks to see if any user-specified parameters are input to the wrapper in the form of a second input argument. If this is not the case, an empty struct is created to hold parameter specifications. Lines 29-36 then populate all non-user specified parameter fields with default values. These lines should largely go unmodified; all MATLAB-based saliency models in SMILER are expected to handle parameters in a similar way, and these lines ensure that the wrapper function communicates with global default specifications in SMILER by `config.json` and model-specific parameters defined in the model's `smiler.json` file.

It should be noted that not all models can be directly integrated into the SMILER ecosystem, however. Some models already include parameters which overlap in functionality or otherwise conflict with the global common parameters that SMILER expects to handle (it is recommended that wrapper authors familiarize themselves with the contents of `[SMILER_root]/config.json`, whose contents are summarized in Table 3.2, in order to see what global parameters are handled by SMILER). In the case of parameter conflicts, new lines should be inserted at Line 41 which detail how to convert SMILER parameters into the expected parameter space of the model. Examples of this can be found in wrapper functions for GBVS and QSS, located in `[SMILER_root]/models/matlab/GBVS/GBVS_wrap.m` and `[SMILER_root]/models/matlab/QSS/QSS_wrap.m`, respectively.

Finally, should any functionality which SMILER expects to be parameterized instead be hard-coded into a model, all efforts should be made to modify the model source code to allow this functionality to be turned off and instead controlled by SMILER when not executing under default parameters. An example of this can be seen in Lines 34-37 of the `[SMILER_root]/models/matlab/CVS/saliencymap.m` file, where automatic conversion of the input image from RGB to CIELAB takes place. This automatic conversion conflicts with SMILER's handling of colour spaces, and thus the conversion call is commented out. A comment is also inserted to document this modification, including the reason for the change in the model source code, the author of that change, and the date when it was applied.

Reading the image (Lines 42-49) is relatively straightforward, and is carried out by a built-in SMILER function `checkImgInput(input_image, params.color_space, enforce3)`, which ensures that the variable `img` which is provided to the saliency calculation function is an image matrix with double format, whether or not `input_image` is specified as a string or image matrix.

The argument `params.color_space` can contain either the string designation for a supported colour space (see Table 3.2) or the string `default`. If the default colour space over which a model operates is not RGB, then lines should be inserted before Line 48 which replace `default` with the designation for the default colour space (see Lines 42-44 of `[SMILER_root]/models/matlab/CVS/CVS_wrap.m` for an example of this).

Finally, should a model require three image channels in order to execute, the `enforce3` argument of the `checkImgInput` function should be set to `true`. If three channels are not required, this argument may either be set to `false` or left off. See Line 45 of `[SMILER_root]/models/matlab/CVS/CVS_wrap.m` for an example of this call when three channels are required.

Calculating the saliency map (Lines 50-58) is the section of the wrapper function which executes a call to the original model source code in order to return a saliency map. Code needs to be inserted at Line 55 which invokes the saliency calculation of the original model source code, as well as provides any model-specific parameters in the format expected by the model source code. The formatting of this code is highly dependent on the manner in which the original source code for the model is implemented, and it is recommended that wrapper authors look at other model wrapper functions for examples. Regardless of the specific form this code takes, a saliency map must be assigned to the `salmap` variable.

The final call of the wrapper function is a call to the SMILER function `fmtOutput`, which interprets and performs any post-processing specified by the parameters, including smoothing, the application of centre bias, and numerical scaling of saliency map values.

Populating `smiler.json` is best done by following the prompts of the `smiler.json` file provided in the `[SMILER_root]/examples/model_skeletons/MATLAB` folder, and shown in Listing 3.8.

Listing 3.8: Contents of the MATLAB template for the `smiler.json` file.

```
1 {
2     "name": "This is where the short name which SMILER will use to refer
       to the model is specified",
3     "long_name": "This is where the full model name is given",
```

```
4       "version": "This specifies the model version within SMILER",
5       "citation": "This provides citation information for the model",
6       "model_type": "matlab",
7       "parameters": {
8           "model_parameter": {
9               "default": "default value",
10              "description": "A description of what this parameter does goes
                  here.",
11              "valid_values": "Either the set of values which the parameter
                  can take or a valid range is specified here."
12          }
13      }
14  }
```

The first several lines contain information which will be used to aid in tracking and citation of the model, and tells SMILER what information should be displayed when calling the `smiler_info` function or executing the `smiler info` command in the CLI. The `name` field contains the SMILER designation of the model, and should match the string used in place of `[MODEL_NAME]` in the wrapper function. The `long_name` field contains the full name of the model. The `version` field should start at 1.0 when a model is first introduced to SMILER. Any major changes to the way the model executes (such as an adjustment of default parameters or an update to the model code to replace deprecated functions) should be accompanied by an incrementation of the model version. This is important for tracking and identifying the source of any potential inconsistencies in future results. The `citation` field should contain full citation information for the model, so that researchers can accurately cite the source material for any models they use in their work.

The final two fields are used by SMILER during model execution. The `model_type` field tells SMILER whether it needs to invoke the MATLAB engine or not in order to execute a model. Since this is a MATLAB model, it should be left as is. The `parameters` field needs to be populated by any model-specific parameters which are available to a user. This field may contain multiple entries, each with the same format. For each parameter, the `model_parameter` string should be replaced by the specific string that will be used to indicate the given parameter. Each parameter is further specified by the contents of three fields: `default`, `description`, and `valid_values`. The `default` field must be populated with the value which will be used in the absence of any user specification. The `description` field should provide a brief and understandable description of what the parameter does and how it affects saliency calculations; this text will be displayed by `smiler_info` calls. The

valid_values field also will be displayed by smiler_info calls, and is used to inform a user what settings are available for the parameter.

3.5.2 Adding Models with Docker

While MATLAB models utilize a common language and engine for execution which makes their addition to SMILER relatively straightforward (as described above), the flexibility of docker invites a greater range of implementation styles and formats which makes it harder to be as specific about implementation details. Nevertheless, the broad pattern of implementation is detailed below.

As with the MATLAB models, a model directory must be created, although in the case of dockerized models the path for this directory is [SMILER_root]/models/docker/[MODEL_NAME], where [SMILER_root] indicates the path to the primary root folder for SMILER and [MODEL_NAME] is the designation by which SMILER will refer to the new model. Within this model directory should go a smiler.json file and a model folder containing the model source code and a run_model.py file.

With dockerized models, the smiler.json file requires several additional information fields, and the contents of the skeleton model file are shown in Listing 3.9. The first several lines are identical to those for a MATLAB file; name, long_name, version, and citation information should be filled out in the same manner as for a MATLAB model. The model_type field has already been set to docker, and so should be left as is.

Listing 3.9: Contents of the template smiler.json for dockerized models.

```
{
    "name": "The short name of the model.",
    "long_name": "The full model name of the model.",
    "version": "Model version within SMILER",
    "citation": "Citation information.",
    "model_type": "docker",
    "model_files": [
        "If there are model prerequisite files (e.g. network weights), put
          them here."
    ],
    "docker_image": "The docker image your model uses. Either built
     locally by you or available on docker hub.",
    "run_command": [
        "python",
        "run_model.py"
    ],
    "shell_command": ["python"],
```

```
16      "parameters": {
17          "model_parameter": {
18              "default": "default value",
19              "description": "A description of what this parameter does.",
20              "valid_values": "Either the set of values which the parameter
                 can take or a valid range is specified here."
21          }
22      }
23  }
```

The field `model_files` should be populated with the names of any external files necessary to run a given model, such as model weights or other pre-defined settings. An example of this can be seen in the `smiler.json` file for the DGII model, found in `[SMILER_root]/models/docker/DGII/smiler.json`. Note that SMILER expects these files to be located in the `[SMILER_root]/models/docker/[MODEL_NAME]/model/` folder.

The `docker_image` field specifies the image container under which the given dockerized model will operate. A number of pre-defined container images are available, either specifically for already existing docker models or via official releases from a number of popular deep learning platforms such as TensorFlow [226], or a model may have a custom image built for it. When building a custom image, it is important that this image be made accessible for other users as well, ideally by hosting that image on Docker Hub [240]. Models with similar requirements may share the same image; an example of this can be seen with the ICF model which utilizes the `tsotsoslab/smiler_dgii` image.

Under most normal circumstances, the `run_command` and `shell_command` fields will remain as they are set, as these should be modified only if a model requires some aspects of normal SMILER functionality to be bypassed. The final necessary information to specify for a dockerized model are the model-specific parameters, which is filled in the same manner as for a MATLAB model.

In addition to the information contained in the model's `smiler.json` file, a `run_model.py` file must be created in the `[SMILER_root]/models/docker/[MODEL_NAME]/model/` folder. This file acts much like the wrapper function for a MATLAB model, and expects as an input argument the path to an input image, and returns as output the corresponding saliency map. Given the variety of ways that dockerized models may be implemented, it is recommended that authors view a number of `run_model.py` files which have been created for models already supported by SMILER in order to see examples of how best to implement this function.

3.6 Discussion

This chapter has presented an overview of the SMILER software package, which provides an open, standardized, and extensible framework for maintaining and executing computational saliency models. The contributions of SMILER are two-fold: a drastic reduction in human effort to set up and run saliency algorithms, and an improvement in the consistency and procedural correctness of results and conclusions produced by different research parties. SMILER is implemented and provided as an open source software project, and it is intended to foster a collaborative research community among researchers interested in exploring computational models of visual salience.

As a continually developing project, it is recommended that users familiarize themselves with SMILER documentation supplied through the GitHub project page to be made aware of any changes or updates not reflected in this document. Researchers are encouraged to contribute their own saliency models to SMILER, and the project includes a set of 'skeleton' models in both MATLAB and dockerized container formats to provide a template and guidance for contributors, as detailed in Section 3.5.

The following chapter provides examples of novel research tasks enabled by the use of SMILER. These are explorations into the performance and behaviour of saliency algorithms of scientific interest to the saliency community, but far enough outside standard benchmarking approaches that each particular study would be difficult to perform without the aid of a software package like SMILER.

Chapter 4

Experiments Enabled by SMILER

The experiments described in this chapter make use of the software described in Chapter 3. The work described here is novel and independently implemented for the purposes of this dissertation.

By leveraging the SMILER library (described in Chapter 3), it is possible to design and run saliency experiments far more quickly than would otherwise be available. This chapter presents a series of novel experiments that are of interest to the saliency research community, but which are far enough outside the bounds of traditional approaches to model benchmarking and comparison that they would likely remain neglected without the ease of use provided by SMILER. Section 4.1 presents a test of saliency model performance on singleton target detection using oriented bar stimuli. Section 4.2 explores saliency model performance on singleton target selection over a series of stimuli that elicit asymmetric search performance in human subjects. Section 4.3 extends the spatially binned analysis of fixation prediction presented in Chapter 2 to the temporal domain, uncovering some interesting spatiotemporal patterns of correlation between human fixations and saliency scores. Overall concluding remarks for all three experiments is provided in Section 4.4.

It should be noted that while SMILER aids in conducting these experiments by drastically simplifying the effort required to generate saliency maps over a dataset, each experiment consists of a novel approach to saliency map analysis that necessitates additional effort outside of the scope of the SMILER project (such as the development of novel metrics and their accompanying calculation code). The details of these extensions and the analysis of their corresponding results will be the focus of this chapter; the saliency maps used in each of these experiments have been calculated using SMILER.

4.1 Saliency Over Oriented Bar Stimuli

The first experiment demonstrating the application of SMILER to a novel search question takes the form of a systematic performance evaluation of saliency model performance over psychophysical search arrays containing oriented bars.

4.1.1 Motivation

Although some commonly used fixation datasets include a small number of artificial stimuli of the sort frequently used in psychophysical research, the primary focus is on natural images. When psychophysical images are included, there is rarely a systematic basis for their generation, and the method of comparison is still either based on fixation prediction or qualitative. However, as

discussed in Section 1.2, many of the foundational assumptions and claims regarding the role of saliency extend beyond fixation prediction. In order to begin probing these properties of saliency algorithm and determining whether or not they meet the claims of human fidelity that are being made, the broad literature of psychophysical research into the characteristics of human vision can be drawn upon. In fact, Bruce *et al.* [157] argue that this body of psychophysical literature ought to serve as the basis for the development of an axiomatic set of performance properties that any saliency model which claims biological fidelity ought to possess.

Oriented bars provide a natural starting point for an investigation into the performance of saliency models for detecting singleton targets. Orientation has been identified as a "basic feature" for visual search [15]. Wolfe provides a list of such features, and defines them as visual attributes that can, given sufficient target-distractor difference, elicit pop-out for a singleton target. Basic feature search (and pop-out in particular) is of particular interest to saliency as it represents one of the motivating observations for the earliest formulations of the saliency map concept [21, 20, 25]. There is a substantial body of literature establishing differences along a basic feature dimension as de facto examples of salient stimuli (*e.g.* [101, 53, 241]), and the attentional pull generated may sometimes even be so strong as to interfere with other visual tasks [51]. Likewise, orientation is either explicitly (*e.g.* the IKN and ICF models) or implicitly (*e.g.* the AIM model) encoded in some manner in the vast majority of saliency models, thereby providing a straightforward and fair test for model coverage of basic psychophysical phenomena. Given the fact that orientation has widespread representation in both human psychophysical literature and model feature sets, if a saliency model fails to identify singleton targets that are salient along the orientation dimension in a manner that is consistent with human performance, it suggests a fundamental lapse in that model's biological fidelity.

4.1.2 Method

4.1.2.1 Performance Evaluation

Determining how best to compare model performance to that of human subjects is not entirely straightforward, of course. The majority of psychophysical studies that look at human behaviour over singleton search arrays have concentrated on response time (RT) rather than fixation prediction

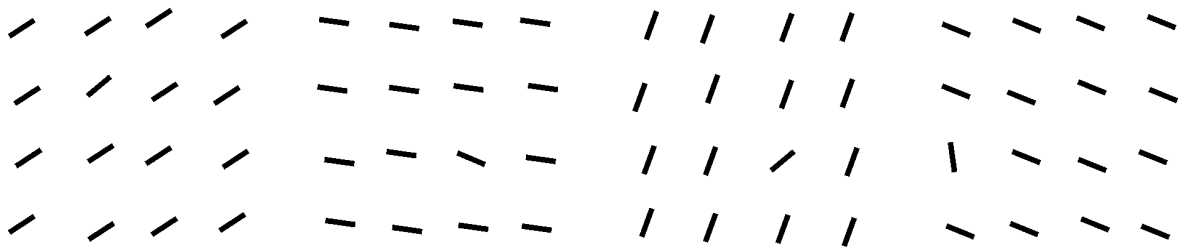
(the focus of most saliency metrics accepted as standard evaluation tools). Even when fixations are tracked, they do not necessarily fully represent the internal representation of saliency. Motter and Belky [242] show in monkeys searching over a search array defined by orientation and colour that a non-negligable number of saccades land in blank portions of the search array not occupied by an stimulus elements. Klein and Farrell [243] have likewise shown that human subjects are able to perform some visual search tasks with fixed gaze with performance comparable to gaze free conditions.

Additionally, when considering metrics for evaluating saliency model fixation prediction, whether it is a Winner-Take-All (WTA) selection strategy that would explicitly simulate fixation points in an image [50] or the metrics discussed in Section 2.2, the majority of approaches (with the exception of NSS and some distribution-based metrics) place a much greater emphasis on rank-order differences across a saliency map rather than comparing the specific saliency value of scene elements. The implicit assumption appears to be that it is not an absolute measure of salience that is important, but rather the relative saliency of one point against its local competitors also vying for attention. Although from the perspective of saliency as an element in gaze control (see Section 1.2.2.4) this appears to be a reasonable assumption given that the eyes must always have a fixation target so long as they are open, it cannot be ruled out a priori that the absolute salience of an object might not also provide important information for human eye movement behaviour. Even for stimuli with the same relative rank order in a saliency map, a larger absolute difference in the salience of targets might hypothetically lend an exploratory urgency to rapidly inspect the high value scene elements, or to a different duration of inhibition of return for previously fixated locations.

There is some psychophysical evidence that, if one wishes to predict visual search dynamics rather than just fixation locations, rank order is not sufficient. Duncan and Humphreys [244] show that either making targets and distractors more similar to each other (increasing target-distractor similarity) or increasing the variety of distractors in a display will increase subject response time in identifying the presence of a unique target. Although Duncan and Humphreys did not frame their analysis of this result from the perspective of saliency, subsequent research by Töllner *et al.* [241] examined both the behavioural performance and neurophysiological correlates of an attentional shift for differing levels of target salience across colour and orientation. They not only replicated Duncan and Humphrey's pattern of increasing RT with increasing target-distractor similarity, but they also

found a commensurate shift in the timing of EEG signals characteristic of an attentional shift. Additionally, Zeheleitner *et al.* [245] found that parametrically manipulating distractor salience would interfere with visual search performance, strongly suggest that relative salience is highly important for response time.

Thus, although the vast majority of saliency models largely ignore the temporal dynamics of visual attention, it is not unreasonable to extrapolate dynamic performance prediction from the relative distribution of saliency values across a scene. An advantage to using a basic feature difference such as orientation to define a salient target is that there is a clear dimension along which the magnitude of that difference may be varied; for oriented bar stimuli, magnitude can easily be measured as the difference in angular tilt between a target bar and a set of distractor bars (see Figure 4.1 for example images). Arun [6] produced a well-characterized performance curve for human response time to an orientation singleton over a broad range of target-distractor orientation differences ranging from $7-60°$. Arun's results are consistent with the results of Duncan and Humphreys: beyond a difference threshold, as the orientation difference between target and distractor decreases, response time increases.



(a) 7° difference **(b)** 15° difference **(c)** 30° difference **(d)** 60° difference

Figure 4.1: Several examples of search arrays with the target at varying levels of orientation difference between the singleton target and the distractor elements. As can be seen, as the angular difference increases the difficulty of search decreases.

By using the same experimental displays as Arun, it is possible to evaluate saliency algorithm performance against two criteria:

1. Does the saliency algorithm consistently find the singleton bar to be the most salient element of the visual scene?

2. Does the ratio of target to distractor salience change as the difference in target and distractor

orientation changes in a manner consistent with human RT changes?

The first test is a gross measure of how well a model is able to generalize to psychophysical stimuli. Performance need not necessarily be perfect. There are a number of potential avenues for numerical errors to occur (such as quantization error), particularly on search arrays with a very small orientation difference. Thus, a model might occasionally highlight a distractor as slightly more salient than a target and still be evaluated as able to accomplish this task. However, ranking the target as less salient than the distractors at any appreciable frequency strongly suggests that a given model strongly deviates from expected human behaviour.

The second test is a more qualitative assessment. Given the general lack of explicit modelling of visual search dynamics by saliency models, there is no clear way to quantitatively convert numerical saliency scores to RT curves. Nevertheless, as discussed above, there is compelling evidence that there is a relationship between RT and the relative salience of target and distractor elements. Therefore, for a given model we should ideally see a pattern of salience ratio between target and distractor bars that is similar to the experimental performance curves exhibited by human observers (Figure 4.2).

When taking the ratio of saliency values between targets and distractors, however, it is necessary to determine how to condense the salience values assigned to all element pixels into a single value. It is unclear what spatial scale is used for integrating saliency signals in the brain in order to capture attention. Taking an average value is less vulnerable to point noise or algorithmic parameter choices such as the size of a smoothing kernel. However, it assumes a mechanism for binding values over an entire element, and averaging over all distractor pixels means that multiple distractor elements are being averaged together which may potentially obscure spatial biases or other sources of heterogeneity when assigning salience values to scene elements. Taking the maximum salience provides a measure that is more in line with standard traversal methods for allocating spatial attention from a saliency map and requires no assumption about object binding, but carries the implicit and unlikely assumption that attentional capture occurs at a pixel-wise spatial scale.

To avoid adopting a potentially misleading set of assumptions, performance was therefore evaluated using both ratio measures: the ratio of the average saliency value of the target against the average saliency value of the distractor set (average-value ratio, or AVR), and the ratio of the max-

imum saliency value for a single pixel on the target against the maximum saliency value for a single pixel from the distractor set (maximum-value ratio, or MVR). Ground-truth masks for the target and distractors were dilated by six pixels to account for the fact that some saliency algorithms that are more edge-based might record the larger saliency values just outside the border of the oriented bars. An error was counted if the ratio between target-distractor saliency values went below 1. For qualitative comparison, Arun's curve from [6] showing human performance as a function of inverted response time plotted against target-distractor orientation difference is provided in Figure 4.2.

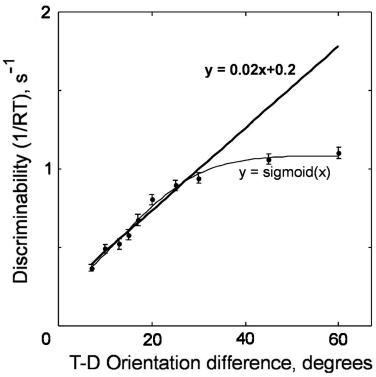


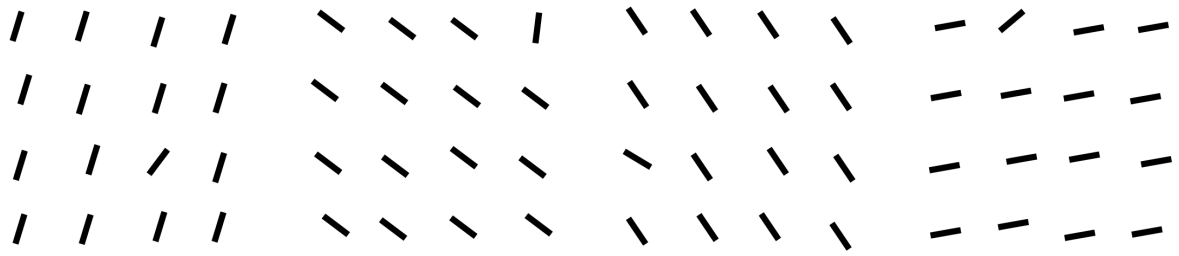
Figure 4.2: Plot of human performance (dark circles) in the form of 1/RT against the target-distractor orientation differences, along with a linear and sigmoidal best fit line. Reproduced from [6].

4.1.2.2 Stimuli

In order to test the performance of saliency algorithms on identifying a singleton target in a search array, a dataset of images was needed. The software tool Psychophysics Image Generator (PIG) [246] was used to generate a set of images with characteristics matching those used by Arun [6]. Each image contains a grid of sixteen oriented bars, arranged in a four by four pattern, with a single target bar and fifteen distractor bars. All distractor bars share an orientation and the target bar differs in orientation by $\pm[7°, 10°, 13°, 15°, 17°, 20°, 25°, 30°, 45°, 60°]$. All other geometric aspects of the stimuli (jitter, image size, bar size, etc.) were matched to Arun's images assuming a pixel-to-degree ratio of 20.

The composition of the dataset was created by generating eight versions of each target-distractor difference (four with a positive difference and four with a negative difference) at each of four possible positions: central, outer corner, outer horizontal, or outer vertical (see Figure 4.3 for examples of

each position). The dataset therefore consisted of 320 total images (8 versions of 10 target-distractor differences at four different position categories).



(a) Central **(b)** Outer Corner **(c)** Outer Horizontal **(d)** Outer Vertical

Figure 4.3: Example psychophysical stimuli generated by PIG showing the four different positions for the target. Each target position can be reflected about the vertical and horizontal midlines, allowing for four equivalent target positions of each category.

4.1.2.3 Models Tested

Saliency maps were calculated for each image using SMILER for the following saliency models: AIM [2], AWS [162], CAS [143], DGII [182], GBVS [78], ICF [184], IKN [25], IMSIG [168], oSALICON [7, 220], SSR [161], and SUN [159]. A YAML file is provided in Appendix A.1 which can be used to generate the saliency maps on which the results of this experiment are based. All algorithms were set to use default parameters and smoothing values, with the exception of the `center_prior` parameter that was set to `none` to attempt to mitigate spatial biases. It is nevertheless clear that some models (such as the publicly released version of the IKN algorithm that was incorporated into SMILER) include undocumented forms of centre bias that are not parametrically controlled. To avoid the risk of unintentially introducing bugs through the modification of a model's source code, these sources of centre bias were left in. It is likely that an implementation of the IKN model that matches the description of the original paper without a centre bias term might perform better on this task. It would also be desirable for the IKN implementation within SMILER to be modified in the future to allow for this bias term to be turned off, but such a change will have to be carefully documented to ensure consistency within the literature.

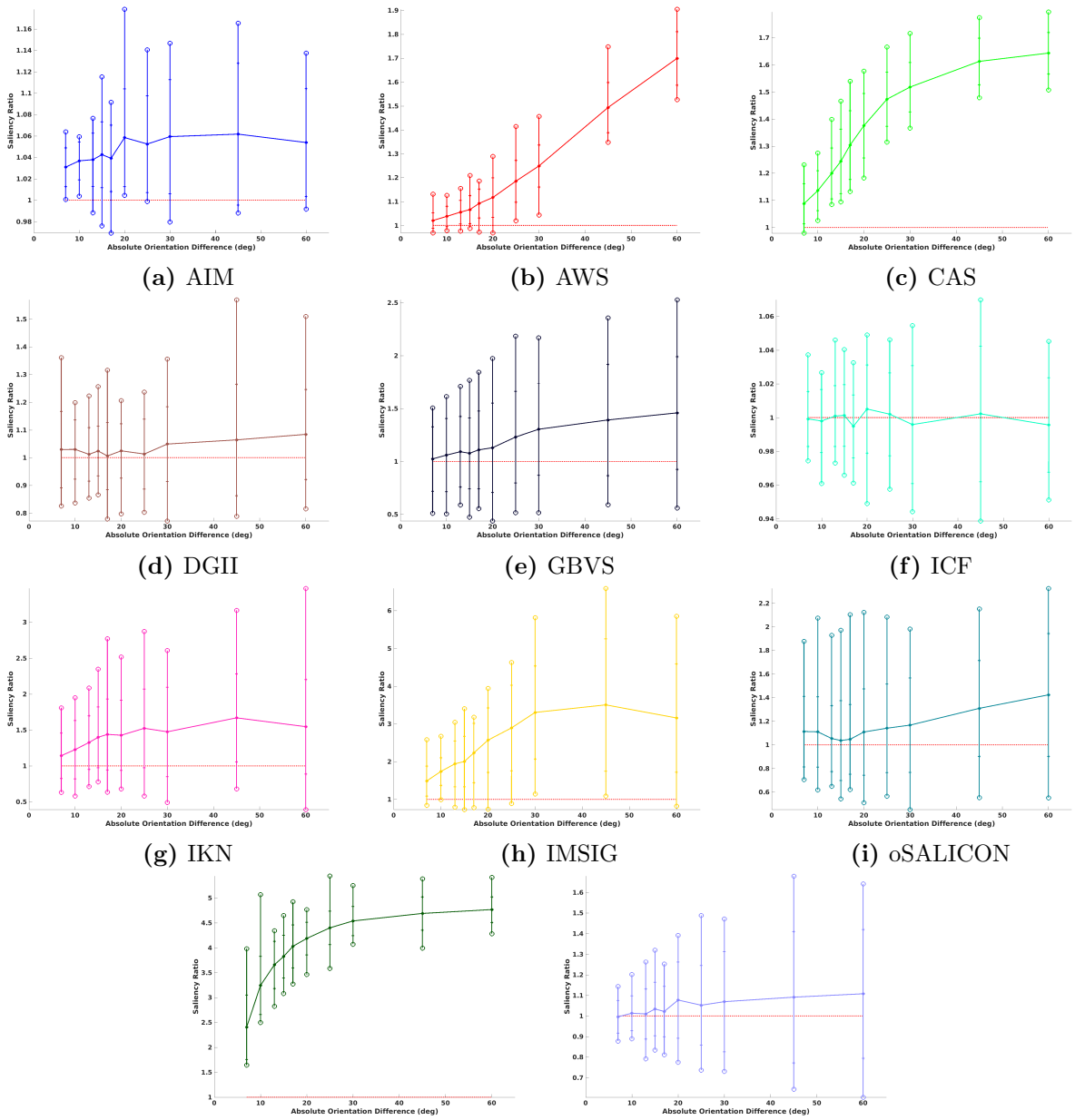
4.1.3 Results

Average target-distractor ratios with respect to orientation difference are shown using the average saliency value (Figure 4.4) and maximum saliency values (Figure 4.5). Error counts are shown in Table 4.1. Interestingly, AIM is the only algorithm tested for which performance gets better by taking the maximum saliency value ratios; all other algorithms perform worse under this test (and sometimes by a rather extreme margin). This may suggest that the standard technique of Winner-Take-All (WTA) selection for sampling fixation points from a saliency map (as in [64], or see Chapter 5 for more discussion of explicit fixation sampling) could be readily improved upon through a more contextual technique of clustering saliency values (possibly through a connected components mechanism to find aggregate pockets of saliency values).

One result of note is that the models based on deep learning (DGII, ICF, and oSALICON) all struggle with this task, suggesting that their ability to predict salient stimuli may not generalize well beyong natural images of the style that they have been trained on. This is perhaps most surprising for ICF, which explicitly does not use a set of learned deep features as input to its readout network but rather makes use of a set of low-level contrast features similar in many ways to more classical models. It therefore explicitly encodes the feature set necessary to perform the task, but has not learned through its readout network a mapping for accurately converting the feature responses to an accurate saliency map for this style of stimuli.

The classical methods that struggle with this task are somewhat more understandable. SUN learns an a priori representation of feature salience that does not respond dynamically to compositional and contextual changes in the stimuli. GBVS has an inherent centre bias that, even when using the "uncentering" parameter, makes it harder for the model to handle stimuli for which the salient target is spatially distributed across the entire visual field. This drawback likewise affects the IKN implementation used here, which includes an undocumented centre bias.

AIM and IMSIG both do a good job of identifying the target consistently with only a small number of errors. However, as the target-distractor orientation difference increases, there is only a small shift in ratio, suggesting that for these two algorithms the identification of salient versus not salient is less graded than for other models. AWS does a better job of grading its response, but the rate of growth of the saliency ratio appears to continuously grow rather than saturate as it



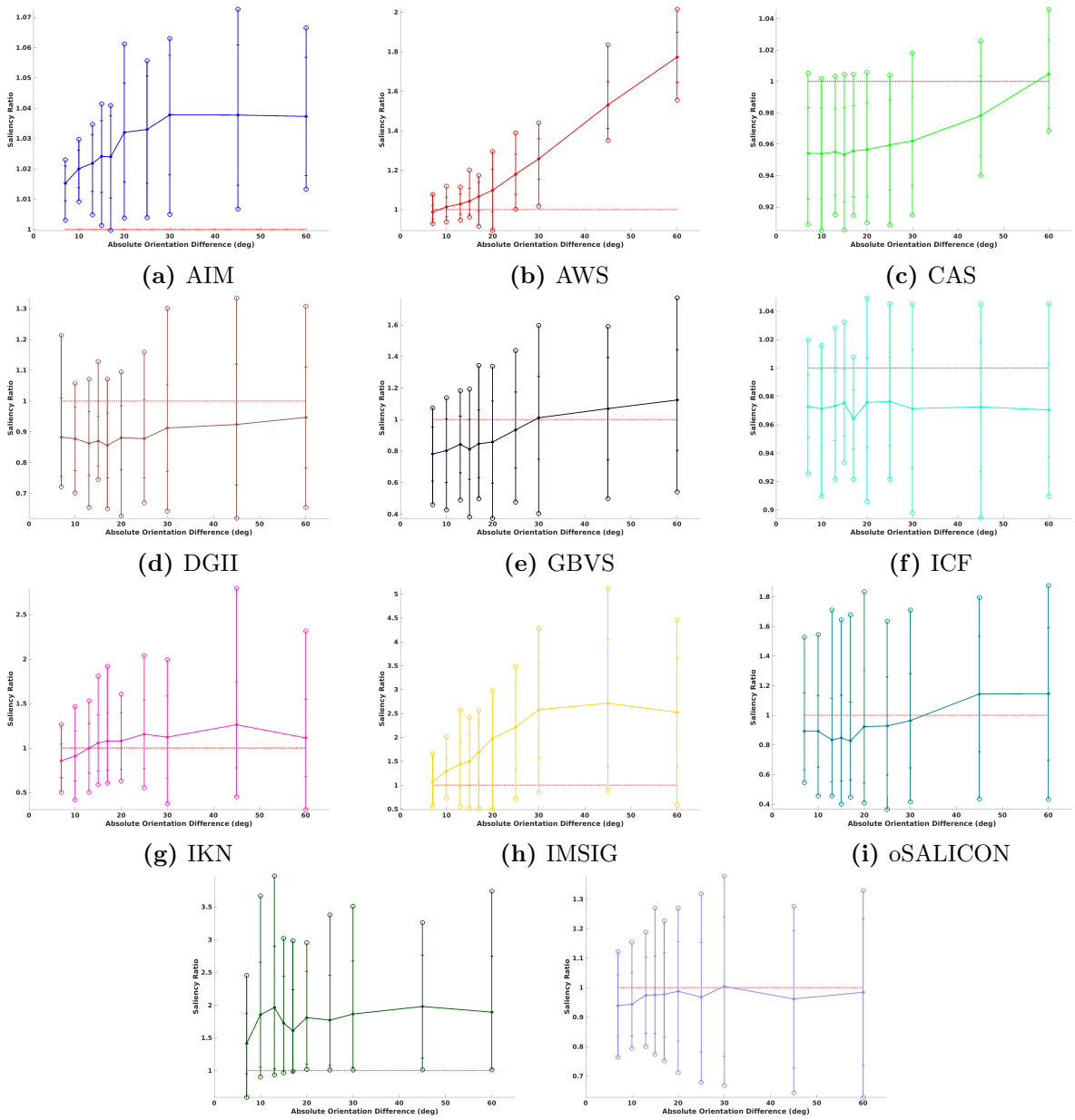
(a) AIM **(b)** AWS **(c)** CAS

(d) DGII **(e)** GBVS **(f)** ICF

(g) IKN **(h)** IMSIG **(i)** oSALICON

(j) SSR **(k)** SUN

Figure 4.4: Plots of target-distractor average saliency ratios (as described in Section 4.1.2.1) for each tested model against target-distractor orientation difference. A ratio of 1 (the error threshold) is shown with a dotted red line. Vertical lines at tested orientation differences show the maximum and minimum ratios recorded (open circles) and the standard deviation of the ratio values (horizontal dashes).



(a) AIM **(b)** AWS **(c)** CAS

(d) DGII **(e)** GBVS **(f)** ICF

(g) IKN **(h)** IMSIG **(i)** oSALICON

(j) SSR **(k)** SUN

Figure 4.5: Plots of target-distractor maximum saliency ratios (as described in Section 4.1.2.1) for each tested model against target-distractor orientation difference. A ratio of 1 (the error threshold) is shown with a dotted red line. Vertical lines at tested orientation differences show the maximum and minimum ratios recorded (open circles) and the standard deviation of the ratio values (horizontal dashes).

Algorithm	AVR Errors	MVR Errors
AIM	32	1
AWS	26	72
CAS	5	267
DGII	140	266
GBVS	126	214
ICF	164	266
IKN	79	152
IMSIG	19	50
oSALICON	120	202
SSR	0	19
SUN	138	186

Table 4.1: Table providing the number of average-value ratio (AVR) and maximum-value ratio (MVR) errors for each algorithm over the dataset of 320 images. An error is counted if the ratio between target salience and distractor salience is below 1.

does for human subjects. When using average saliency ratios, CAS and SSR both produce curves that qualitatively match the psychometric curves recorded for human subjects, and SSR manages to do this without any errors. Interestingly, CAS suffers an enormous drop in performance when only the maximum pixel value ratio is considered, and SSR, although it mostly manages to avoid errors under this scheme, loses the structure of a saturating curve.

4.1.4 Discussion

Although a relatively simple test, the experiment presented here nevertheless highlights an interesting range in algorithm performance over psychophysical stimuli. Of particular note is the fact that the most recently developed algorithms actually have the worst performance. This suggests that while modern deep learning efforts are continuously pushing the limits of performance for fixation prediction on natural image datasets, they may be diverging from the underlying cognitive features and strategies that guide human behaviour. It may be possible to augment model training to include stimuli of this sort to improve performance, but recent work by Kim *et al.* [247] on the same-different task suggests that the sort of relative contextual comparison necessary to render an accurate assessment of saliency over singleton search arrays may be outside the scope of feedforward networks (a class into which the vast majority of deep learning-based saliency models fall).

This result also showcases the usefulness of having easy access to a standardized library for model

execution. As performance benchmarks become more established, there is the risk of overfitting efforts toward performance on a specific dataset under a specific evaluation metric. Without a relatively simple and straightforward way for independent researchers to probe model behaviour, it becomes more difficult to run these sorts of sanity checks that investigate aspects of model performance outside the narrow bounds of a small set of benchmarks.

4.2 Saliency in Asymmetric Search Arrays

A second case study for the deployment of SMILER is provided here in the area of *search asymmetries*. A search asymmetry occurs when it is faster for an observer to find some search target A amongst a set of distractors B than when searching for a target B amongst a set of A distractors. Search asymmetries represent another class of visual search characteristics that are well studied in human psychophysical experiments, but have received only a small amount of attention in the saliency literature.

4.2.1 Motivation

Early investigations concentrated on asymmetries involving basic features, noting that it was easier to identify the presence of a unique feature than the absence of that feature [248, 249], a difference that persists whether or not the target is known ahead of time [250]. For example, even though both targets pop out in both condition, it is nevertheless easier to find a magenta target amongst blue distractors than a blue target amongst magenta distractors. In this case, the magenta target contains both blue and red features, and thus may be detected by the presence of activity on the red channel, whereas a blue target amongst magenta distractors is instead unique due to the absence of red, as both magenta and blue contain the same level of activity over the blue channel. It is important to note that one must also consider both feature heterogeneity and the effect of a search field background when determining the existence of an asymmetry, as failure to do so might lead to a false conclusion of asymmetry [251, 252].

Over time, it has become clear that there are many ways to elicit asymmetries. In addition to it being easier to search for feature presence over feature absence (referred to as *basic asymmetries*), there is also a class of *novelty asymmetries* [253]. Novelty asymmetries refer to search conditions

in which it is easier to find a target with unusual or unexpected characteristics (such as an upside down letter) amongst distractors that are more familiar (such as the same letter rightside up).

The range of stimuli that can exhibit novelty asymmetries appears rather large, from stimuli composed of relatively sparse feature sets, such as letters, through to much more complex shapes, such as the silhouette of an animal [253]. Nevertheless, all novelty asymmetries appear to rely to a degree of stimulus familiarity or meaning. For example, letter asymmetries can disappear when testing subjects familiar with both symbols (for example, searching for a backwards N amongst forward facing N distractors is faster for English speakers than the converse condition, but not for speakers familiar with both the Latin and Cyrrilic alphabets [254]).

Although there is little debate as to the existence of novelty asymmetries, there is some debate about whether or not novelty effectively constitutes a feature that guides visual search [255, 256]. Bruce and Tsotsos [156] argue that the most parsimonious explanation for a novelty asymmetry rests in the interactions of neuronal sparse coding [257] and distractor inhibition. They propose that familiar targets will have been encoded in a manner with sparse representation, allowing their inhibition as scene elements of low interest to be more easily accomplished. Targets that are unfamiliar lack a compact representation, meaning that their inhibition as a distractor tends to either more difficult or less complete. Interestingly, this proposal is consistent with the results of Shen and Reingold [254] who find that search is more efficient when the distractors are familiar, regardless of whether or not the target is familiar, and it echoes the earlier argument of Rauschenberger and Yantis [258] that redundancy in terms of the ease of encoding distractor sets is the primary driver of apparent asymmetries in visual search.

As models of visual saliency have grown more complex either in proscribed features (such as the numerous streams of the RARE2012 algorithm [148]) or, more recently, the reliance on deep networks with a large number of learned features (*e.g.* [186, 182, 189]), there is now a wide spectrum in the richness of encoded feature space on which the saliency signal is calculated. Likewise, the manner in which those encoded features are converted into a saliency map also frequently varies from model to model, and asymmetric visual search provides a tool to investigate whether or not a given approach replicates human behaviour. While improving benchmark performance is often presented as a sign that saliency algorithms are continually improving in their explanation of human fixation selection [259, 260], the results of Section 4.1 suggest that it is not enough to simply

look at performance over natural images, and that many of the most modern saliency algorithms struggle the most with tests over psychophysical stimuli. Search asymmetries are therefore used to extend the results of Section 4.1, as the selection of different sets of asymmetric stimuli allow us to investigate if it is the composition of stimulus features that dictate algorithm performance.

4.2.2 Method

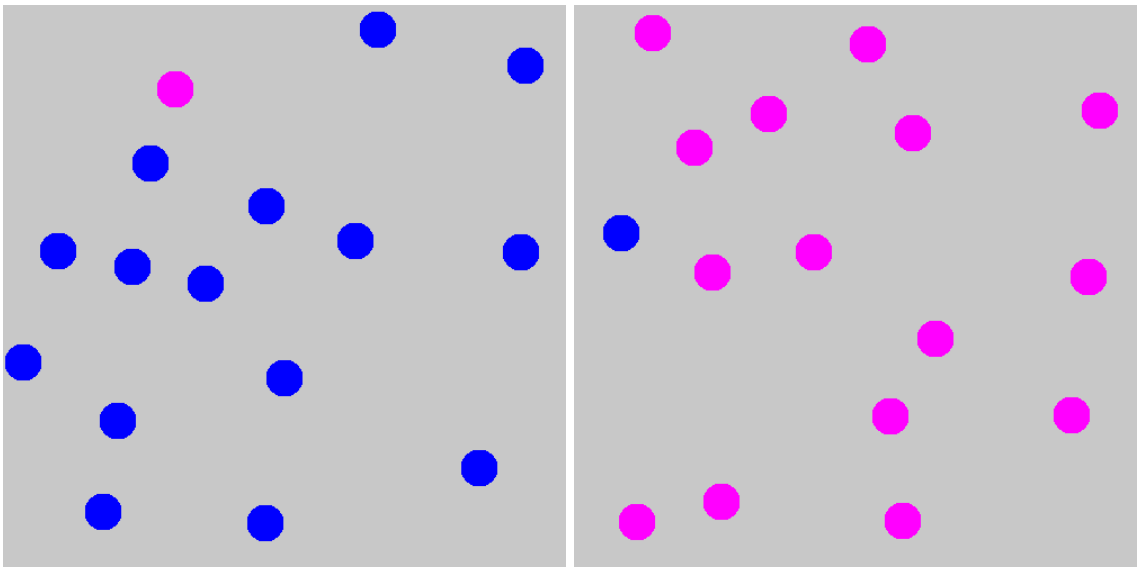
4.2.2.1 Stimuli

As with the orientated bar search task presented in Section 4.1, this experiment required the creation of a custom set of stimuli that would form target-distractor pairs likely to elicit asymmetric search performance in human subjects based on the principles identified in prior human research. Images were composed by the following target-distractor pairs:

1. Classic Asymmetries
    (a) Blue/Magenta dots (see example in Figure 4.6)
    (b) O/Q (see example in Figure 4.7)
    (c) A/flipped-A (see example in Figure 4.8)
    (d) Q/flipped-Q (see example in Figure 4.9)
2. Complex Target Asymmetries (see Figures 4.10-4.12)
    (a) Person/flipped Person

Images based on classic asymmetry pairs provide both basic feature asymmetries (sets of blue and magenta coloured dots and the classic O and Q pair), as well as letter-based novelty asymmetries consisting of a canonically oriented letter and a vertically flipped letter. Classic asymmetry pairs were created using 32×32 pixel patches containing the stimulus elements randomly scattered over a square search array of size 500×500 pixels with a minimum distance between image elements of 32 pixels. Each image contained one target and 15 distractors. In order to reduce the likelihood of an errant result based on interactions with the background, letter images were generated with black letters on a a set of background values: white ($RGB = [255, 255, 255]$, example shown in Figure 4.8), light grey ($RGB = [200, 200, 200]$, example shown in Figure 4.6) and grey ($RGB = [127, 127, 127]$,

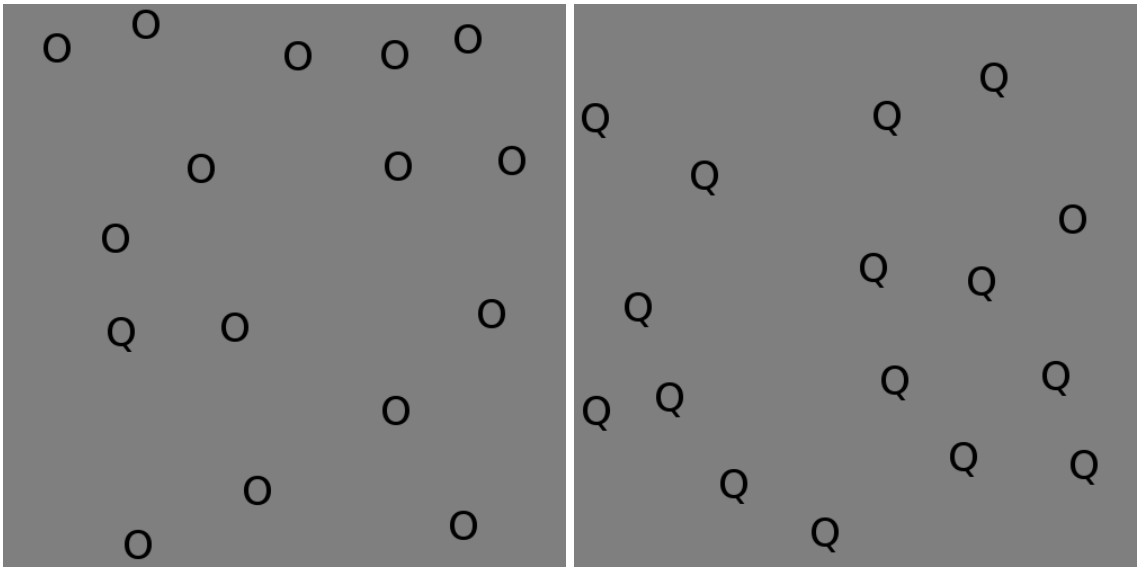
example shown in Figure 4.7). A set of inverted intensity stimuli were also generated, with white letters on a set of background values consisting of: grey ($RGB = [127, 127, 127]$), dark grey ($RGB = [55, 55, 55]$), and black ($RGB = [0, 0, 0]$, example shown in Figure 4.9). Blue/Magenta dot stimuli used the same set of background values (white, light grey, and grey), but never inverted the dot colours. Twenty images for each stimulus set with a given target-distractor designation on each background value, resulting in 240 images for each letter-based asymmetry pair and 120 images for the Blue/Magenta dots.



(a) Target is a blue dot **(b)** Target is a magenta dot

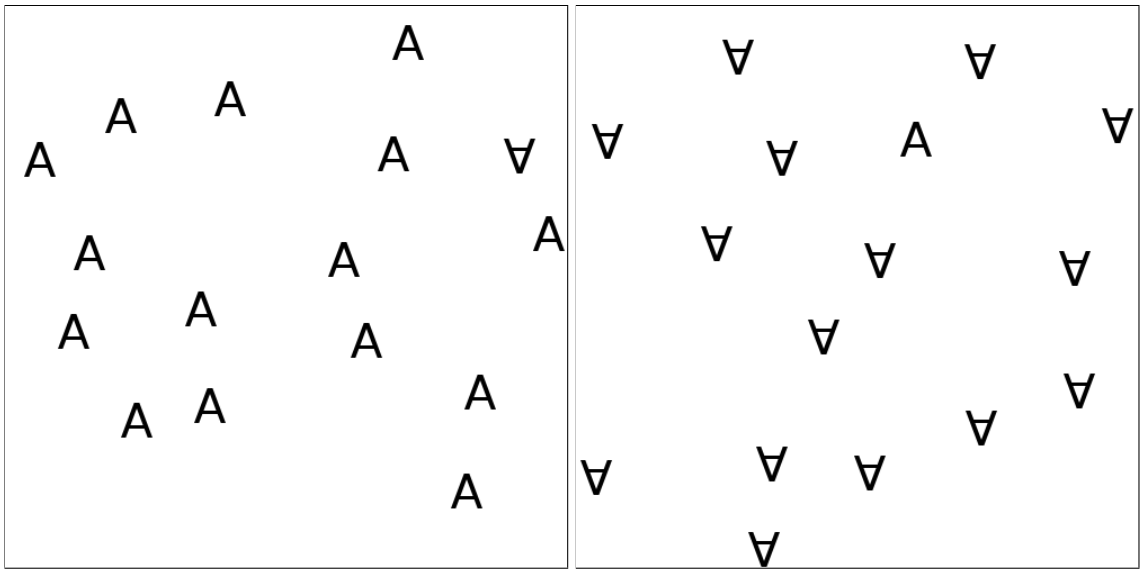
Figure 4.6: Example images from the Blue/Magenta dots stimulus set with a light grey background. The image on the left should result in faster search times for human observers compared to the image on the right.

Complex target asymmetries were set up by extracting a seven exemplar person elements from the MS-COCO dataset [261], a large dataset of labeled natural scenes. MS-COCO was selected due to its inclusion of tight object masks and similarity to the images on which many of the deep learning saliency algorithms were trained to extract features. A set of human elements were extracted from the image by randomly selecting candidate images and then retaining the ones in which an object could be discerned by its silhouette (in order to allow for an asymmetry experiment in which a flipped silhouette target is presented amongst canonically oriented silhouettes. This design was originally presented with an example involving elephants, leading to the moniker "dead elephant"



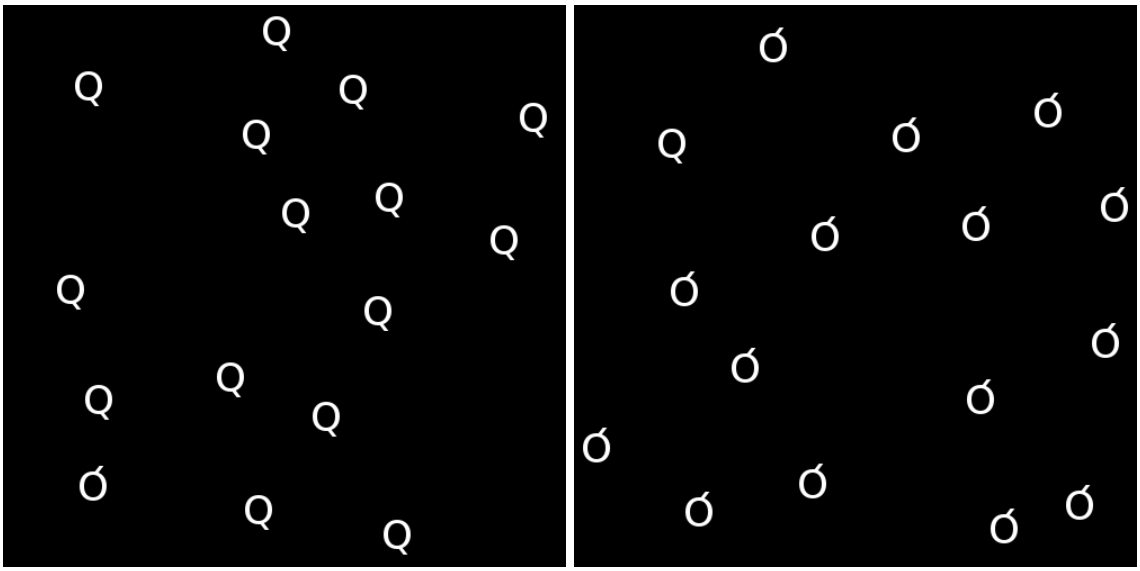
(a) Target is a Q **(b)** Target is an O

Figure 4.7: Example images from the O/Q stimulus set with a grey background. The image on the left should result in faster search times for human observers compared to the image on the right.



(a) Target is a flipped-A **(b)** Target is a canonical A

Figure 4.8: Example images from the A/flipped-A stimulus set with a white background. The image on the left contains a novel target amongst familiar distractors, and should therefore result in faster search times for human observers familiar with the Latin alphabet compared to the image on the right.



(a) Target is a flipped-Q **(b)** Target is a canonical Q

Figure 4.9: Example images from the Q/flipped-Q stimulus set with white letters on a black background. The image on the left contains a novel target amongst familiar distractors, and should therefore result in faster search times for human observers familiar with the Latin alphabet compared to the image on the right.

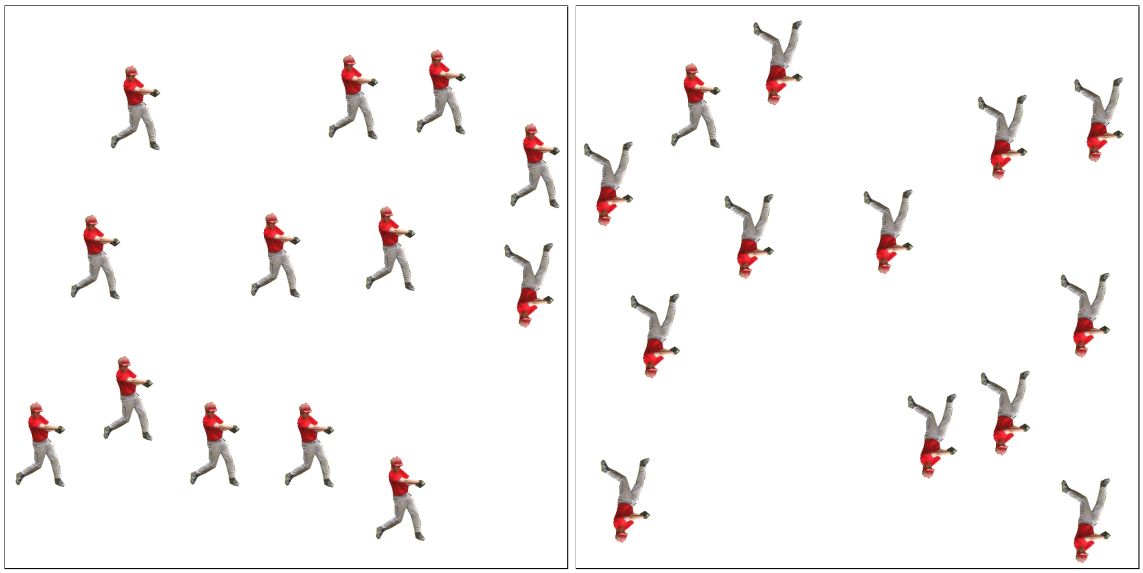
[253]). The extracted elements are shown in Figure 4.10. Note that when producing the search arrays, all elements were resized such that the major axis (in this case, height) of the element image was set to 150 pixels. Person elements were chosen because of the high degree of representation humans have in most major machine learning datasets, as well as the importance placed on human targets by deep learning based saliency algorithms.

(a) Person 1 **(b)** Person 2 **(c)** Person 3 **(d)** Person 4 **(e)** Person 5 **(f)** Person 6 **(g)** Person 7

Figure 4.10: Person exemplars extracted from the MS-COCO dataset

All complex target-distractor pairs were composed of either a canonically oriented target amongst

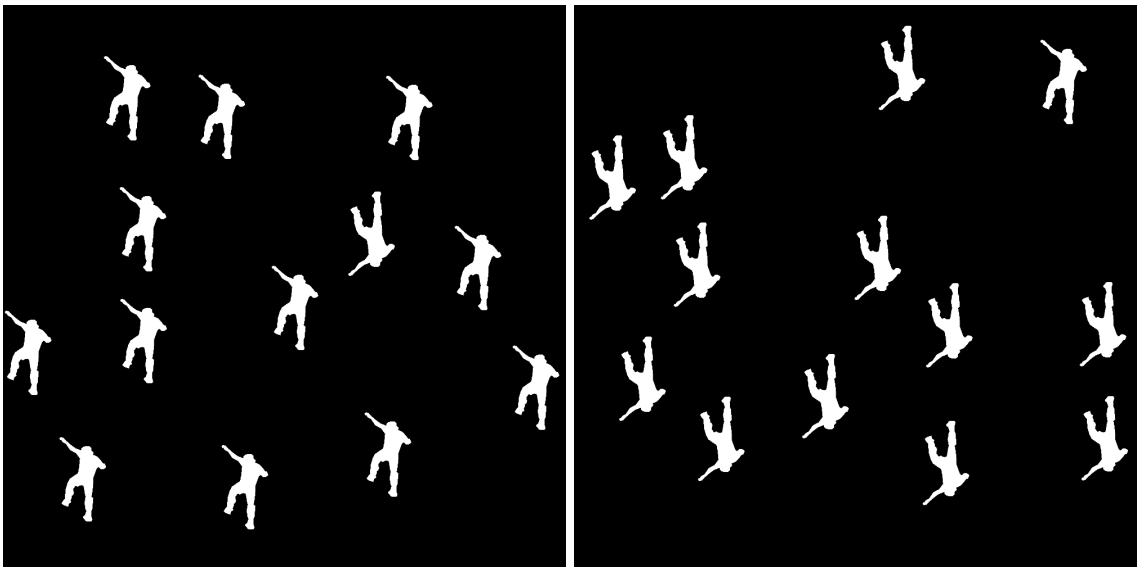
flipped distractors, or the converse. Images were composed of target-distractor pairs using the fully detailed elements themselves (see Figure 4.11 for an example) as well as the element masks (see Figure 4.12 for an example). In order to accomodate the larger and more complex stimulus elements in the complex scenes, the image canvas was increased to a size of 1000×1000 pixels, the minimum distance between image elements was set to 40 pixels, and the number of distractors was reduced to 12. Background values were set to be white, grey, or black, with objects placed on each background and black object masks placed on the white and grey background, and white object masks on the grey and black backgrounds. 10 images were generated for each target-distractor setting on each background value, resulting in a total of 140 images for each exemplar.



(a) Target is a flipped Person **(b)** Target is a canonical Person

Figure 4.11: Example images with Person 4. The image on the left has an unusual orientation and should therefore lead to faster average search times than the image on the right.

It is important to acknowledge that the stimuli composed by complex objects is specifically chosen to conform with the types of images used in the training of machine learning based saliency algorithms. Although these images have not explicitly been tested for an asymmetry in response time with human observers, they are nevertheless formulated to be consistent with previously seen asymmetries. The mask-based images can be seen to emulate the "dead elephant" search demonstrated by Wolfe [253], and at least some evidence exists to suggest that the results for schematized artificial stimuli extends to more ecologically natural stimuli. For example, Horstman



(a) Target is a flipped Person 1 mask **(b)** Target is a canonical Person 1 mask

Figure 4.12: Example images with the Person 1 mask. The image on the left has an unusual orientation and should therefore lead to faster average search times than the image on the right.

and Bauland [262] show that search asymmetries that are elicited in sparse drawings of faces based on the apparent facial emotion displayed also extend to more natural depictions of faces. Ballaz *et al*. [263] likewise showed that for detailed depictions of animals, non-canonical orientations of the animal were asymmetrically found in comparison to upright targets. The stimuli designed for this experiment should therefore be consistent in formulation with search arrays that have previously been shown to elicit search asymmetries, while retaining features that should be familiar to the learned network models.

4.2.2.2 Models Tested

Saliency maps were calculated for each image using SMILER for the following saliency models: AIM [2], AWS [162], CAS [143], CVS [145], DGII [182], DVA [160], GBVS [78], ICF [184], IKN [25], IMSIG [168], QSS [167], RARE2012 [148], SalGAN [189], oSALICON [7, 220], SAM-VGG [186], and SSR [161]. YAML files are provided in Appendix A.2 which can be used to generate the saliency maps on which the results of this experiment are based. All algorithms were set to use default parameters and smoothing values, with the exception of the `center_prior` parameter that was set to `none` to attempt to mitigate spatial biases. As with the experiment presented in Section 4.1, it

should be noted that this was not able to completely eliminate the spatial bias of all models, but rather only those for which the spatial bias was explicitly applied an accessible manner which could be excised.

4.2.2.3 Performance Evaluation

As with the oriented bar experiment of Section 4.1, evaluation of saliency algorithm performance for this experiment is not based on eye movement prediction, but rather on the evaluation of relative salience between target and distractor objects. This evaluation is measured with three quantities: the ratio of average target saliency to average distractor saliency, the ratio of maximum target salience to maximum distractor salience, and the number of steps to the target in a rank order traversal. Both ratio measurements are calculated in an identical manner to the performance calculations in Section 4.1, though in this case there is no systematic variation in search difficulty and ratios are therefore found as single values for each target-distractor pair.

The addition of the steps to target metric in this study is not without some necessary caveats. While cumulative probability of fixations or number of fixations to target is a common performance measure for visual search tasks that also include eye tracking (*e.g.* [242, 264, 265]), this is not directly analogous here. As discussed in Section 4.1.2.1, fixations in visual search tasks may not land directly on stimulus elements [242], and the singleton may sometimes be found in the absence of eye movements entirely [243]. Much of the challenge of linking fixations to saliency maps rests on the fact that saliency models for fixation prediction are modelling an active, dynamic process through a static map. This topic is explored in more detail in Chapter 5.

Therefore, while steps to target resembles the number of fixations assessment of performance employed by research using human and monkey observers, it should not be viewed as a direct comparison to human fixation patterns. Rather, it is simply intended to provide a more nuanced exploration of scene element rank ordering and rate of target acquisition predicted by the saliency model. It was felt that this additional analysis was necessary given the increased complexity of the stimuli used in this experiment compared to the oriented bar stimuli tested in Section 4.1. Some specific patterns may be clearly viewed as errors (such as failing to find the target completely) or extremely unlikely to reflect human behaviour (such as fixating the target only after every distractor has been examined), but other performance patterns (such as fixating the target first

on every trial, versus a more gradual accumulation of target acquisition rate) are not necessarily meant as a representation of relative performance between models.

The number of steps to the target is calculated by assuming a rank-order traversal of the saliency map, applying at each step an inhibition of return (IOR) region to cover the detected target. Each IOR region is a rectangle equal to double the size of a search element (this way it should fully cover a search element even if a fixation occurs near the edge of an element). Up to 21 fixations were allowed before a time-out was triggered and the model was judged to have missed the target in that image. The size and shape of this IOR region was not chosen for any biological fidelity, but rather to be as generous as possible to the algorithms being evaluated. Nevertheless, as will be seen, this is a task over which many models struggle greatly. This measure is converted to a cumulative probability of target acquisition given x steps by totaling the number of trials for a given condition for which the target was aquired by a given model in x or fewer steps.

4.2.3 Results

This section provides a visualization of algorithm performance according to the three metrics described in Section 4.2.2.3: line plots showing the cumulative probability of fixating the target using rank-order traversal, and box plots showing target-distractor ratio of average saliency values and target-distractor ratio of maximum saliency values. Figures displaying results for each metric are split into two groups of algorithms based on alphabetical ordering (AIM-ICF and IKN-SSR). Boxplots display a colour-filled box that extends from the lower to upper quartile of the data with a central line at the median. Whiskers extend from the box to show the range of the data according to the equations:

$$W_{low} = \arg\min_d (d > Q_1 - 1.5R) \tag{4.1}$$

$$W_{high} = \arg\max_d (d < Q_3 + 1.5R) \tag{4.2}$$

where W_{low} is the lowest extent of the whiskers, W_{high} is the highest extent of the whiskers, Q_i is the ith quartile value, d is a specific datum from the data, and R is the interquartile range $Q_3 - Q_1$. Outliers beyond this range are plotted as open circles.

For each plot showing the results for a given metric, two boxplots are shown representing each target-distractor arrangement for a given stimulus pair. The target for a given result is labelled on

the x-axis. For each condition the target for which human observers are expected to have increased performance in locating is shown on the right in pink, while the more difficult target condition is shown on the left in blue. For the plots showing maximum or average saliency ratios, a green line is drawn at a ratio of 1; values below this line indicate that a particular model finds targets to be less salient than distractors for the given condition.

Effect sizes and statistical significance are summarized in Tables Table 4.3, 4.4, 4.6 and 4.7. Effect size for average saliency is calculated as the change in average saliency ratios, Δ_{avg}, between the condition Target A among Distractors B and the condition Target B among Distractors A, calculated as

$$\Delta_{avg} = \frac{1}{N}\sum_{t \in T} a_t^B - a_t^A \tag{4.3}$$

where t is a trial, T is the set of all trials for a given experimental condition, a_t^A is the average saliency of target A in trial t, and a_t^B is the average saliency of target B in trial t, and N is the number of trials for a given condition. Effect size for maximum saliency, Δ_{max}, is calculated as

$$\Delta_{max} = \frac{1}{N}\sum_{t \in T} x_t^B - x_t^A \tag{4.4}$$

where x_t^A is the maximum saliency ratio of target A in trial t, and x_t^B is the maximum saliency of target B in trial t. Note that this calculation is expected to return a positive value if a given model accurately reflects human behaviour. Statistical significance is calcualted using Welch's t-test [266] and included in parentheses. Entries which achieve standard levels of significance are noted by italicized ($p < 0.05$) and bold ($p < 0.01$) fonts.

4.2.3.1 Classic Asymmetries

This section presents results from a number of classic asymmetry search formulations, including two commonly framed as "feature presence/absence" asymmetries (magenta versus blue dots and Q's versus O's), as well as two "novelty" asymmetries based on letter orientation (flipped and canonically oriented A's and Q's). Table 4.2 presents the percentage of trials for which targets were not found after 21 steps for each target/distractor condition, while Tables 4.3 and 4.4 present the average differences in average and maximum saliency ratios, respectively, between the paired

Target/Distractor	AIM	AWS	CAS	CVS	DGII	DVA	GBVS	ICF
Blue/Magenta Dots	0	0	0	0	1.7	6.7	3.3	0
Magenta/Blue Dots	0	0	0	0	38.3	0	15.0	95.0
O/Q	0	0	0	3.3	15.0	0.8	15.0	16.7
Q/O	0	0.8	0	2.5	23.3	0	14.2	0
A/flipped A	0	0	0	0.8	3.3	0	10.8	2.5
flipped A/A	0	0	0	5.0	14.2	0	11.7	0.8
Q/flipped Q	0	0.8	0	3.3	22.5	0.8	9.2	2.5
flipped Q/Q	0	0	0	0.8	20.8	0	9.2	5.0

Target/Distractor	IKN	IMSIG	QSS	RARE2012	SalGAN	oSALICON	SAM	SSR
Blue/Magenta Dots	28.3	0	0	-	10.0	5.0	15.0	0
Magenta/Blue Dots	26.7	11.7	0	-	43.3	5.0	11.7	6.7
O/Q	25.0	0	-	0	28.3	1.7	20.8	9.2
Q/O	25.0	0	-	0	24.2	0	35.0	4.2
A/flipped A	19.2	0	-	0	25.8	0	33.3	0
flipped A/A	20.0	0	-	0	23.3	0	22.5	0
Q/flipped Q	20.8	0	-	0	18.3	0.8	30.0	7.5
flipped Q/Q	19.2	0	-	0	28.3	0	25.0	2.5

Table 4.2: Table showing the percentage of missed targets for each target-distractor condition. Note that RARE2012 is missing results for the Blue and Magenta dots due to processing errors on those stimuli, and QSS is missing results on the letter-based stimuli due to processing artifacts that appear to affect black and white stimuli.

asymmetric target conditions. Specific patterns of results for each target/distractor set of conditions are discussed below. Note that results for QSS are only presented for the conditions containing blue and magenta dots, as black and white stimuli appear to elicit an unexpected artifact consisting of an intensely bright patch in the upper left corner. Due to this clear error in output, it was decided that QSS would be disqualified for these inputs.

Target A	**Target B**	**AIM**	**AWS**	**CAS**	**CVS**
Blue Dot	Magenta Dot	***-0.362 (0.00)***	0.856 (0.88)	***-1.305 (0.00)***	***-180.460 (0.00)***
O	Q	-0.001 (0.91)	***0.386 (0.00)***	0.005 (0.47)	***-0.550 (0.00)***
A	flipped A	-0.013 (0.06)	-0.005 (0.45)	-0.002 (0.69)	-0.090 (0.51)
Q	flipped Q	0.014 (0.06)	0.010 (0.28)	0.009 (0.20)	-0.026 (0.87)
Target A	**Target B**	**DGII**	**DVA**	**GBVS**	**ICF**
Blue Dot	Magenta Dot	***-1.715 (0.00)***	***23.020 (0.00)***	***-0.670 (0.00)***	***-10.094 (0.00)***
O	Q	-0.001 (0.91)	0.060 (0.09)	0.112 (0.12)	*0.442 (0.04)*
A	flipped A	0.144 (0.09)	-0.001 (0.96)	-0.017 (0.80)	0.058 (0.23)
Q	flipped Q	0.627 (0.22)	0.042 (0.13)	0.057 (0.42)	-0.158 (0.06)
Target A	**Target B**	**IKN**	**IMSIG**	**QSS**	**RARE2012**
Blue Dot	Magenta Dot	-0.110 (0.22)	***-1.962 (0.00)***	***2.635 (0.00)***	-
O	Q	*0.192 (0.01)*	***0.380 (0.00)***	-	***0.170 (0.00)***
A	flipped A	-0.026 (0.67)	-0.013 (0.37)	-	0.042 (0.05)
Q	flipped Q	0.061 (0.41)	-0.007 (0.63)	-	0.032 (0.15)
Target A	**Target B**	**SalGAN**	**oSALICON**	**SAM**	**SSR**
Blue Dot	Magenta Dot	***-0.652 (0.00)***	***-0.151 (0.00)***	0.165 (0.15)	***-0.344 (0.00)***
O	Q	*0.176 (0.03)*	***0.447 (0.00)***	***-0.223 (0.01)***	*0.121 (0.01)*
A	flipped A	0.019 (0.78)	-0.032 (0.20)	***0.572 (0.00)***	0.068 (0.13)
Q	flipped Q	-0.144 (0.07)	***-0.385 (0.00)***	0.131 (0.14)	-0.018 (0.68)

Table 4.3: Table showing the change in average saliency values assigned to Target B in comparison to Target A (p-value shown in parentheses). Results with $p < 0.05$ are italicized, and results with $p < 0.01$ are written in bold. Note that RARE2012 is missing results for the Blue and Magenta dots due to processing errors on those stimuli, and QSS is missing results on the letter-based stimuli due to processing artifacts that appear to affect black and white stimuli.

Target A	**Target B**	**AIM**	**AWS**	**CAS**	**CVS**
Blue Dot	Magenta Dot	***-0.369 (0.00)***	***-3.185 (0.00)***	***-0.265 (0.00)***	***-38.946 (0.00)***
O	Q	***0.014 (0.00)***	***0.457 (0.00)***	***0.050 (0.00)***	*-0.213 (0.01)*
A	flipped A	***0.013 (0.00)***	-0.003 (0.58)	-0.001 (0.85)	0.002 (0.97)
Q	flipped Q	***0.019 (0.00)***	0.011 (0.29)	0.005 (0.35)	-0.001 (0.99)
Target A	**Target B**	**DGII**	**DVA**	**GBVS**	**ICF**
Blue Dot	Magenta Dot	***-0.804 (0.00)***	***21.048 (0.00)***	***-0.394 (0.00)***	***-4.538 (0.00)***
O	Q	0.044 (0.44)	***0.136 (0.00)***	***0.103 (0.01)***	0.352 (0.05)
A	flipped A	*0.146 (0.03)*	-0.004 (0.83)	-0.012 (0.75)	0.033 (0.43)
Q	flipped Q	0.308 (0.20)	0.016 (0.23)	0.012 (0.74)	-0.081 (0.05)
Target A	**Target B**	**IKN**	**IMSIG**	**QSS**	**RARE2012**
Blue Dot	Magenta Dot	-0.038 (0.46)	***-1.705 (0.00)***	***2.334 (0.00)***	-
O	Q	***0.140 (0.00)***	***0.292 (0.00)***	-	***0.174 (0.00)***
A	flipped A	0.008 (0.83)	-0.016 (0.26)	-	***0.069 (0.00)***
Q	flipped Q	0.025 (0.51)	*0.025 (0.04)*	-	0.023 (0.31)
Target A	**Target B**	**SalGAN**	**oSALICON**	**SAM**	**SSR**
Blue Dot	Magenta Dot	***-0.299 (0.00)***	*-0.061 (0.02)*	***0.172 (0.01)***	-0.003 (0.83)
O	Q	0.062 (0.09)	***0.364 (0.00)***	-0.076 (0.07)	***0.064 (0.00)***
A	flipped A	0.020 (0.57)	0.028 (0.14)	***0.183 (0.00)***	0.013 (0.14)
Q	flipped Q	*-0.073 (0.04)*	***-0.388 (0.00)***	0.032 (0.41)	0.005 (0.77)

Table 4.4: Table showing the change in maximum saliency values assigned to Target B in comparison to Target A (p-value shown in parentheses). Results with $p < 0.05$ are italicized, and results with $p < 0.01$ are written in bold. Note that RARE2012 is missing results for the Blue and Magenta dots due to processing errors on those stimuli, and QSS is missing results on the letter-based stimuli due to processing artifacts that appear to affect black and white stimuli.

4.2.3.1.1 Blue Dot vs. Magenta Dot: The results for visual search arrays formed by blue and magenta dots (examples shown in Figure 4.6) are presented in Figures 4.13-4.18. Figures 4.13 and 4.14 show the cumulative probability of finding the target as the number of steps increases. Figures 4.15 and 4.16 show the average saliency ratio, while the maximum saliency ratio is shown in Figures 4.17 and 4.18. Note that results for RARE2012 are missing for this category; when blue and magenta dots were presented over a white background, the fact that the blue channel of the image was fully saturated over all pixels appears to cause the RARE2012 algorithm to fail.

It is interesting to note that more than half of the tested algorithms show an asymmetry in the direction opposite what is expected of human observers. AIM, CAS, CVS, and IMSIG all showed a significant preference for and high performance in the blue target condition but still exhibited at least passable performance in the magenta condition. CAS and CVS were able to consistently find the colour singleton on the first step, regardless of the target condition, and represented the asymmetry solely in the saliency ratio values. AIM and IMSIG always find the blue target on the first step, whereas the magenta target is only found first a small fraction of the time (but is nevertheless always found on subsequent steps for the AIM model).

GBVS, oSALICON, and SSR also showed a preference for the blue target (significant in average ratio, and also significant in maximum ratio for GBVS) but only moderate performance overall in both conditions; the target was rarely located early, but by the final allowed step the target had usually been found. DGII and SalGAN similarly both show significant preference for blue targets, but exhibit only moderate performance which drastically degrades on magenta targets (missing nearly half of all magenta targets). ICF shows perhaps the most extreme asymmetry in performance, with very high performance when finding the blue target, but completely failing to detect the magenta target on all but a minute fraction of trials. It is unclear why this deficit in performance occurs for the magenta condition; it could be that the colour space ICF operates over (which is based on the principle components computed over the MIT1003 dataset [4]) provides poor representation for the magenta condition, or it could be an issue with the fixation data used to train the network (this data is the same as used to train DGII, which suggests it may in part be due to this issue given DGII's poor performance on magenta, but the fact that DGII succeeds for some magenta trials suggests that there must be other factors, too).

In contrast, DVA and QSS showed a relatively strong and significant preference for the magenta

target, perhaps to the point of extremes; DVA struggles markedly when presented with a blue target. In fact, DVA distinctly persists in highlighting magenta distractors as more salient than a singleton blue target, showcasing a lack of flexibility in representation (probably the most extreme case aside from the aforementioned performance of ICF). SAM's performance is somewhat interesting; it is the only deep learning algorithm to show a preference for magenta targets, albeit a preference that is relatively small in magnitude (though significant in maximum ratio). SAM's performance overall is only moderate, with a non-negligible number of misses in both target conditions. IKN is notable in that it showed very little asymmetry, but overall did not perform particularly well at this task.

The performance of AWS is somewhat unusual. While it does not show a significant asymmetry toward blue targets in maximum ratio, in both target conditions it appears to preferentially find the distractors. As seen in Figure 4.13, for the vast majority of trials AWS successfully finds the target only on the sixteenth step after first already investigating every distractor.

Overall, QSS is the only algorithm to replicate expected performance for this experimental condition. QSS consistently finds the target first, calculates the target salience as higher than the distractors, and finds a higher target to distractor salience ratio (for both maximum and average values) when the target is magenta amongst blue than in the converse condition.

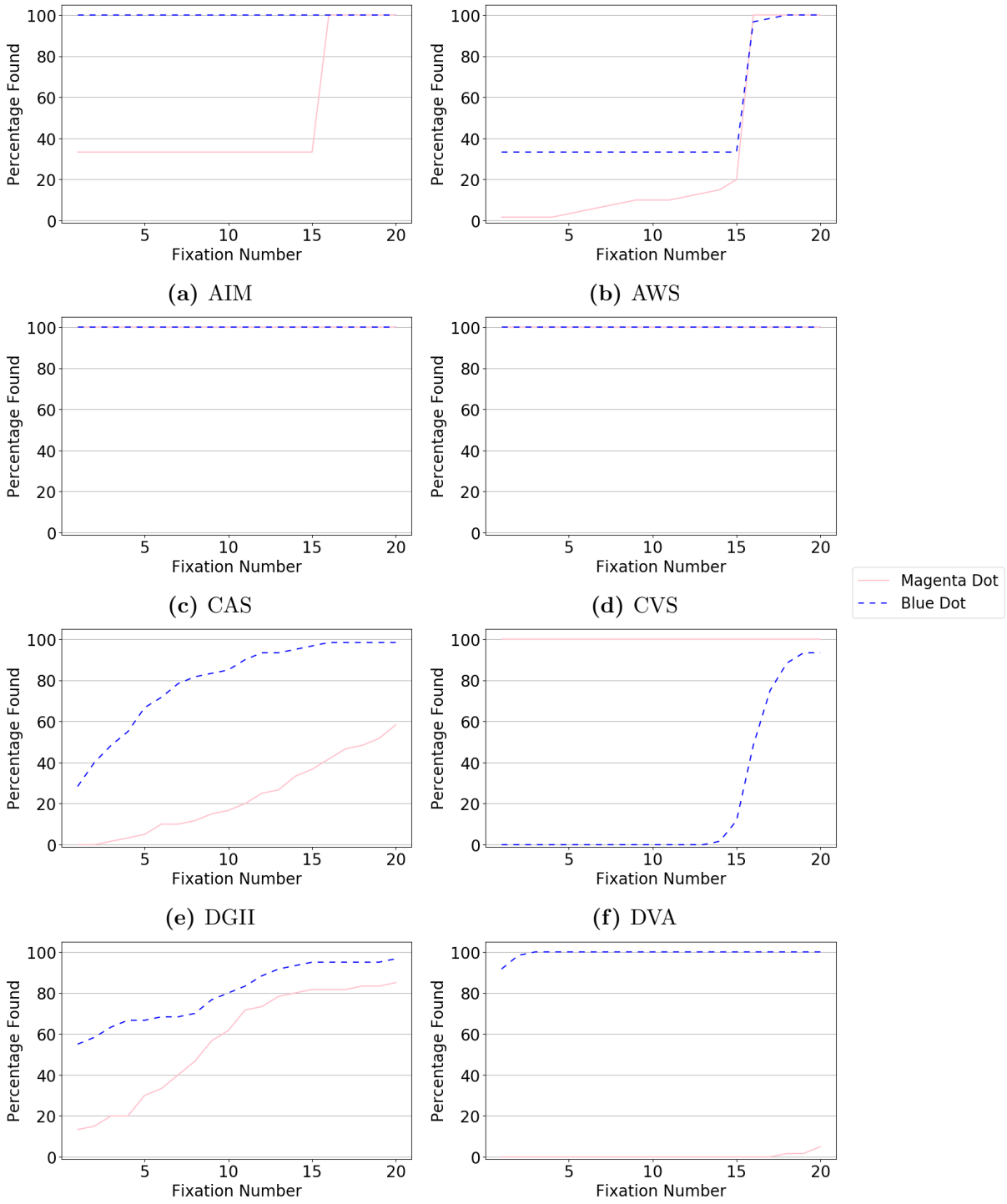
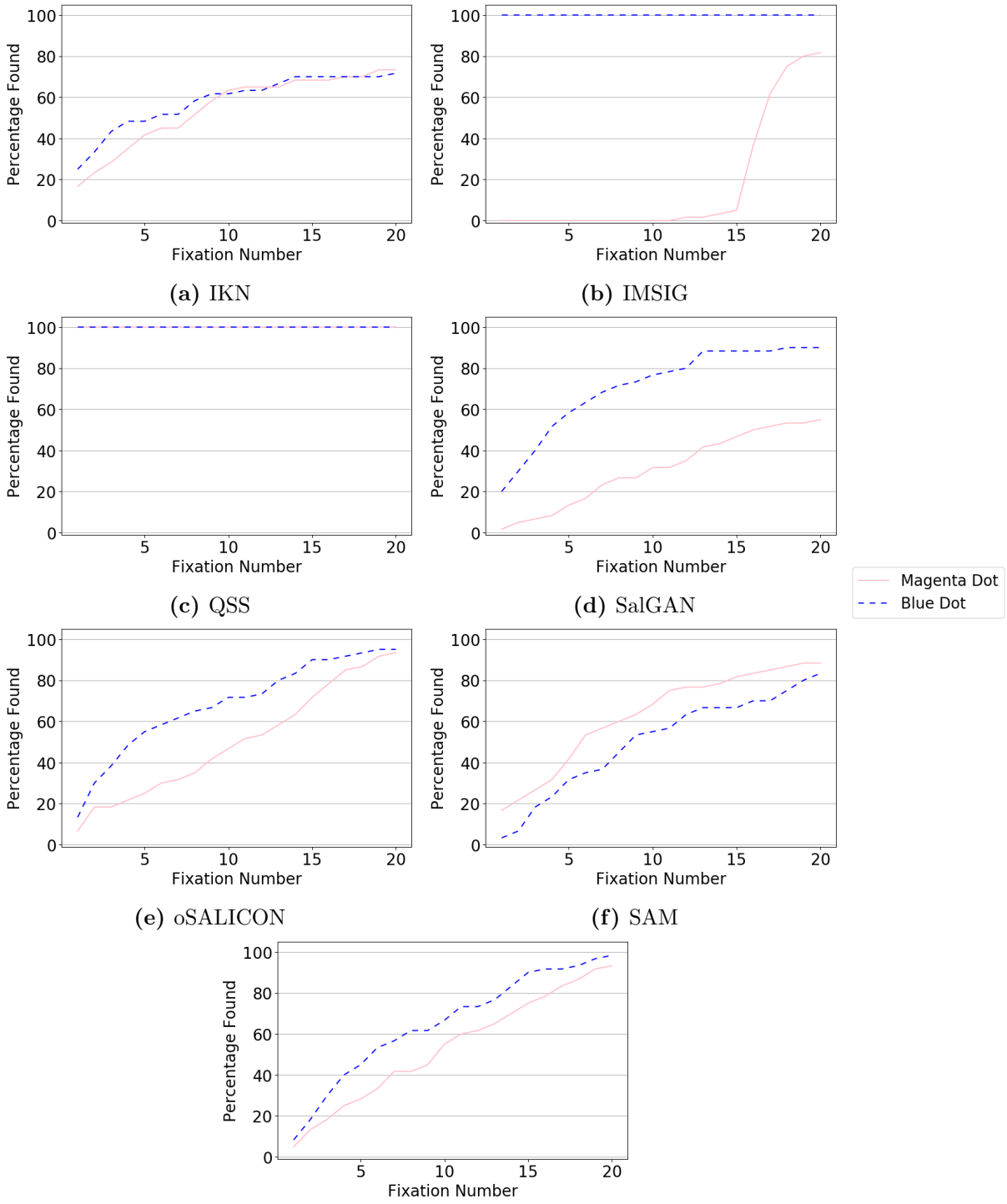
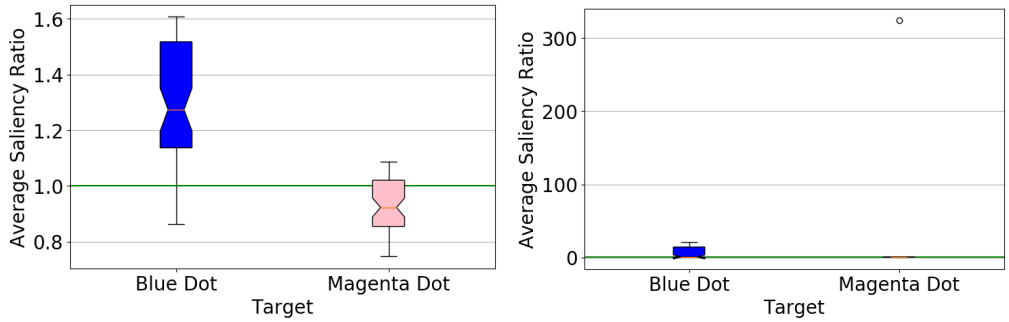


Figure 4.13: The cumulative probability of finding the target for a given number of stes for models AIM-ICF when searching for a magenta target amongst blue distractors (pink line) or when searching for a blue target amongst magenta distractors (dashed blue line).



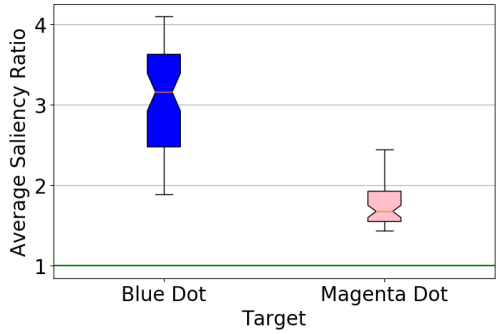
(a) IKN **(b)** IMSIG **(c)** QSS **(d)** SalGAN **(e)** oSALICON **(f)** SAM **(g)** SSR

Figure 4.14: The cumulative probability of fixating the target for a given number of steps for models IKN-SSR when searching for a magenta target amongst blue distractors (pink line) or when searching for a blue target amongst magenta distractors (dashed blue line).

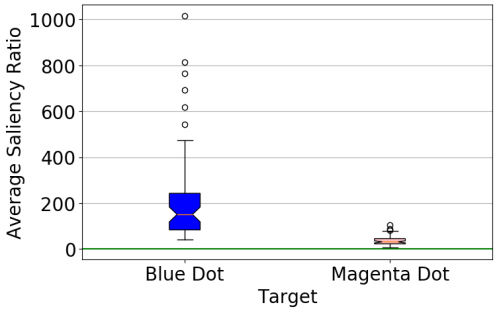


(a) AIM

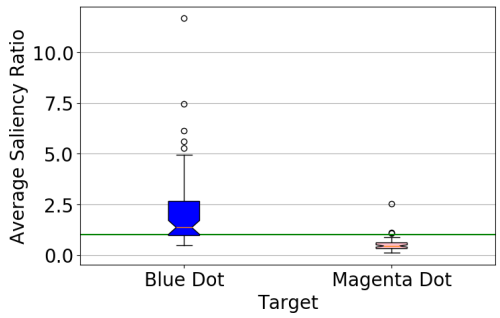
(b) AWS



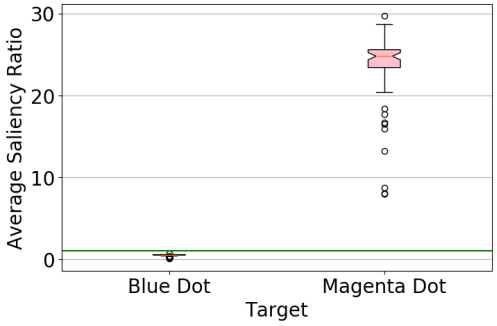
(c) CAS



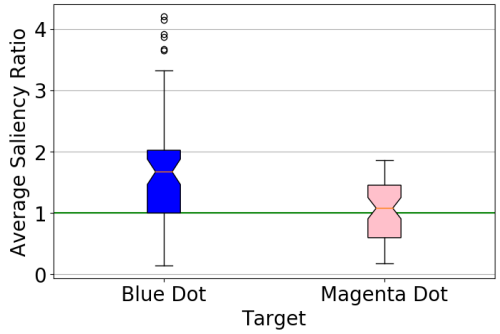
(d) CVS



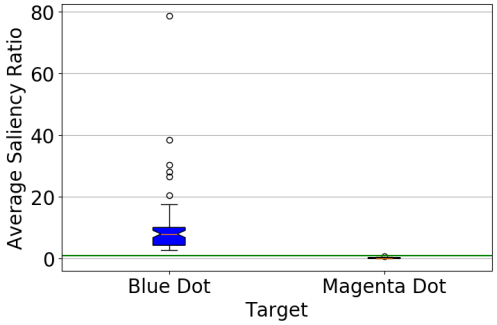
(e) DGII



(f) DVA

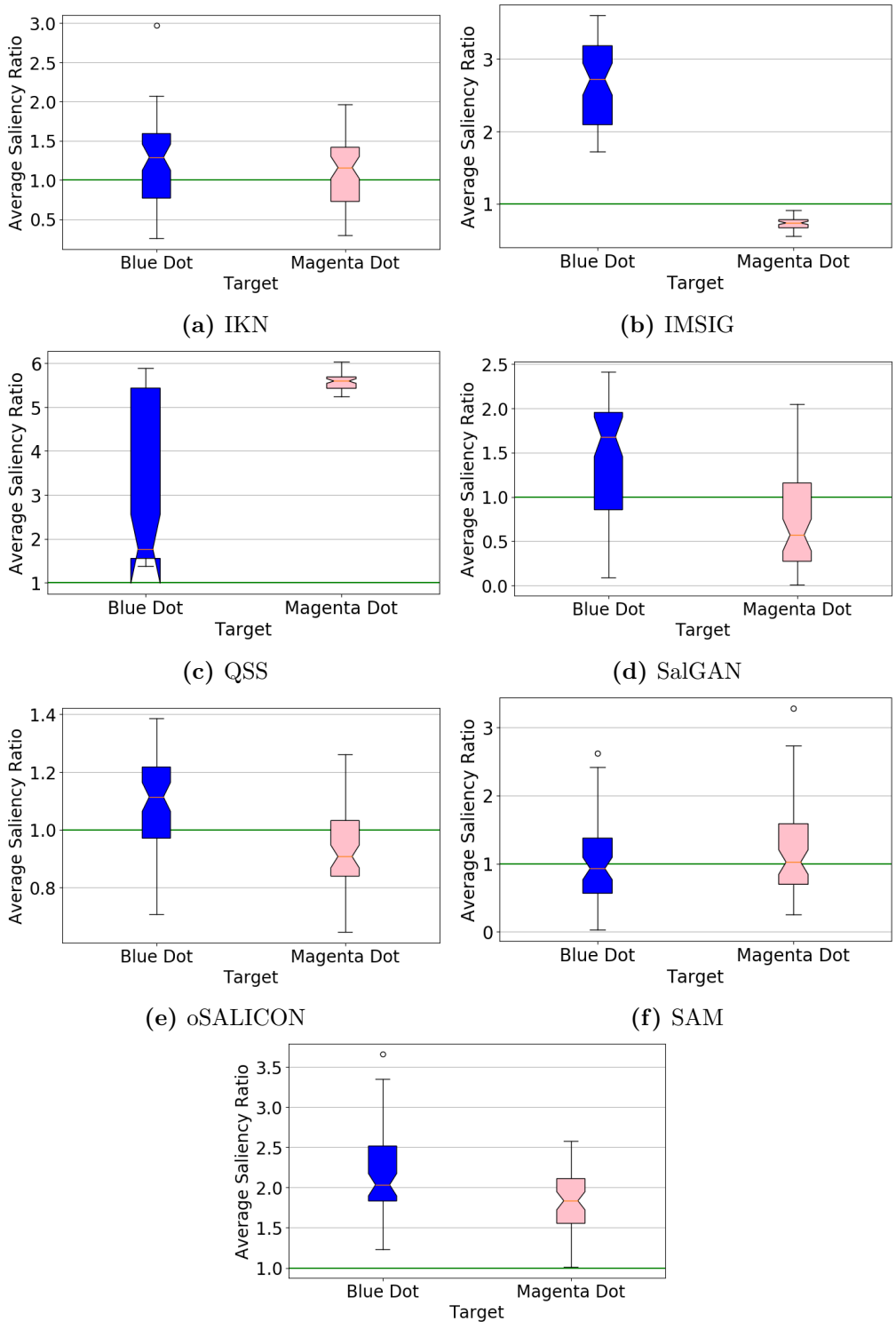


(g) GBVS



(h) ICF

Figure 4.15: The ratio of target to distractor average saliency values for AIM through ICF. Results on the left in each plot (blue) are for blue targets amongst magenta distractors, whereas results on the right (pink) are for magenta targets amongst blue distractors.



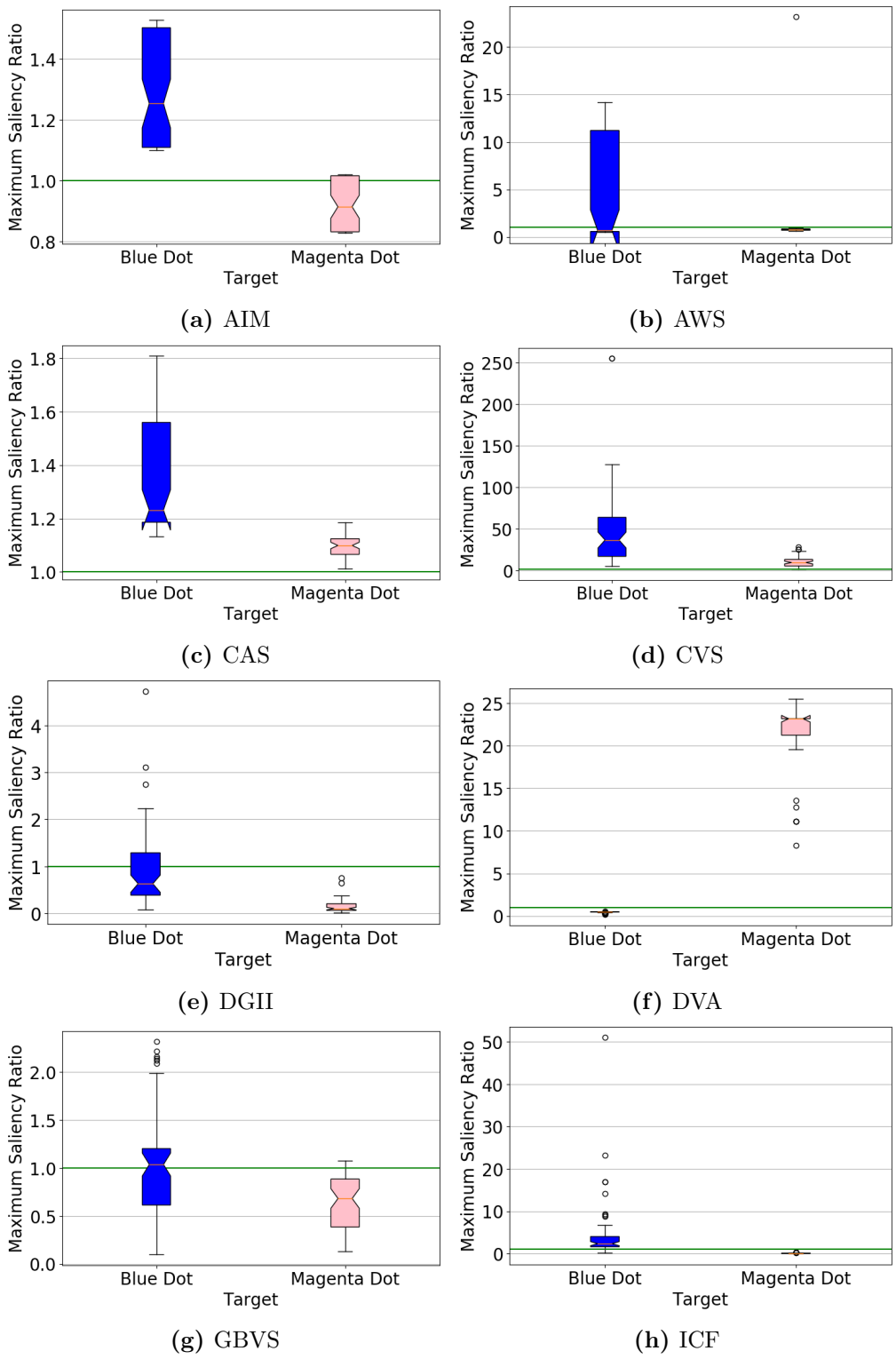
(a) IKN **(b)** IMSIG

(c) QSS **(d)** SalGAN

(e) oSALICON **(f)** SAM

(g) SSR

Figure 4.16: The ratio of target to distractor average saliency values for IKN through SSR. Results on the left in each plot (blue) are for blue targets amongst magenta distractors, whereas results on the right (pink) are for magenta targets amongst blue distractors.



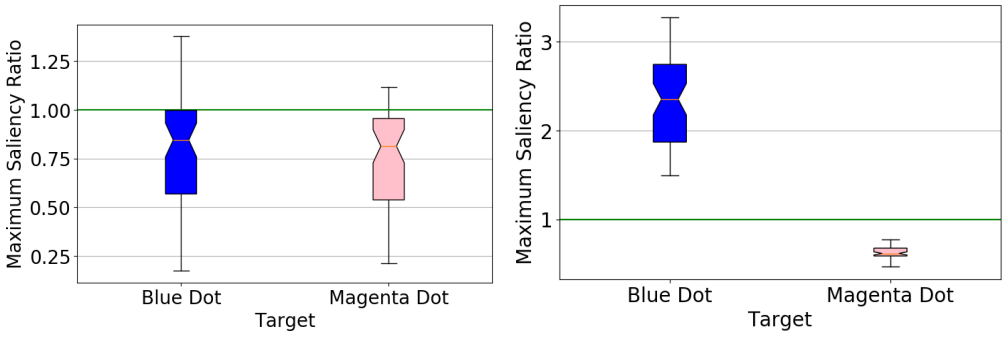
(a) AIM **(b)** AWS

(c) CAS **(d)** CVS

(e) DGII **(f)** DVA

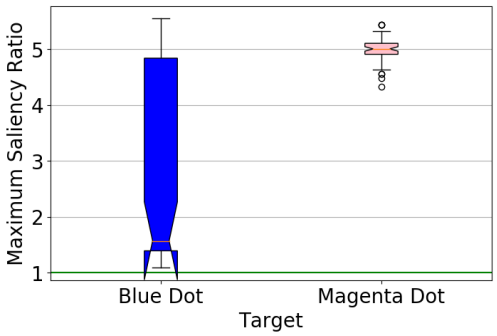
(g) GBVS **(h)** ICF

Figure 4.17: The ratio of target to distractor maximum saliency values for AIM through ICF. Results on the left in each plot (blue) are for blue targets amongst magenta distractors, whereas results on the right (pink) are for magenta targets amongst blue distractors.

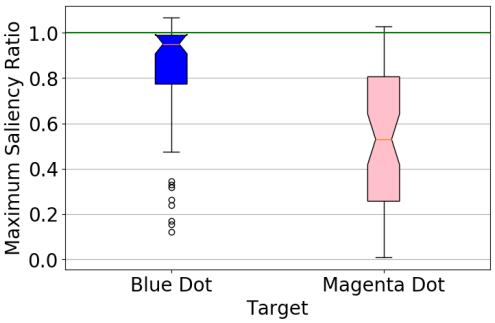


(a) IKN

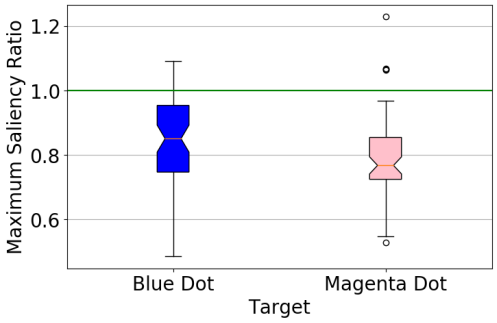
(b) IMSIG



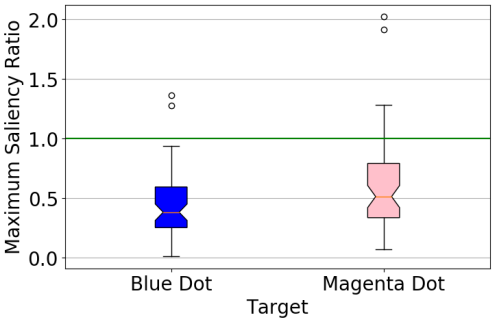
(c) QSS



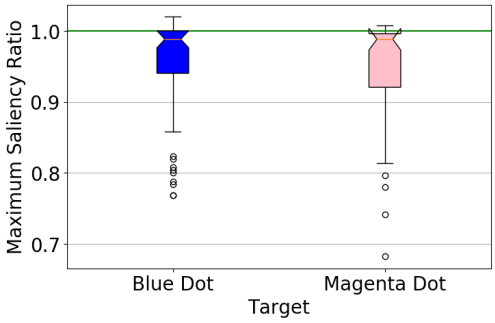
(d) SalGAN



(e) oSALICON



(f) SAM



(g) SSR

Figure 4.18: The ratio of target to distractor maximum saliency values for IKN through SSR. Results on the left in each plot (blue) are for blue targets amongst magenta distractors, whereas results on the right (pink) are for magenta targets amongst blue distractors.

4.2.3.1.2 Q vs O: The results for visual search arrays formed by Q and O letters (examples shown in Figure 4.7) are presented in Figures 4.19-4.24. Figures 4.19 and 4.20 show the cumulative probability of fixating the target as the number of steps increases. Figures 4.21 and 4.22 show the average saliency ratio, while the maximum saliency ratio is shown in Figures 4.23 and 4.24.

Q and O arrays form the second and last of the asymmetry conditions tested here which are based on an asymmetry classically interpreted as a presence/absence asymmetry along a basic feature channel (red activity in the case of a magenta dot amongst blue dots, and an oriented bar intersecting the circle in the case of a Q amongs O's). This asymmetry should therefore be reasonably expected to be reflected by any tested algorithm. In contrast to the coloured dot experiment, this condition has been shown to be more difficult than straight pop-out, and thus finding the target first is not necessarily the most reasonable expected outcome. However, one would reasonably expect that in either condition the target should still be found, and for humans typically is found using fewer fixations than the number of elements in the search array [106].

It is unfortunate that QSS, which performed admirably in the magenta and blue experimental condition, had to be discounted in this condition due to what appears to be an error with black and white stimuli; all maps were dominated by an extremely bright square in the upper left corner, with only minor numerical differences present across the rest of the map. While unlike the issue with RARE2012 for the blue and magenta dots in which the model failed to produce output, in this case QSS did provide output which could still be analyzed. However, it was felt that the extreme nature of this artifact was sufficient in magnitude to be disqualifying.

The vast majority of algorithms tested exhibit improved performance when searching for a Q target amongst O distractors. For most algorithms this is a marked improvement in target acquisition rate, though for a small number the difference is not exceedingly large. GBVS, IKN, SalGAN, and SSR struggled with this condition for both targets, and only had a slightly higher rate of target location and saliency ratios when the target was a Q rather than an O. DGII, by contrast, showed no clear preference for either target type, and did relatively poorly in its rate of target acquisition.

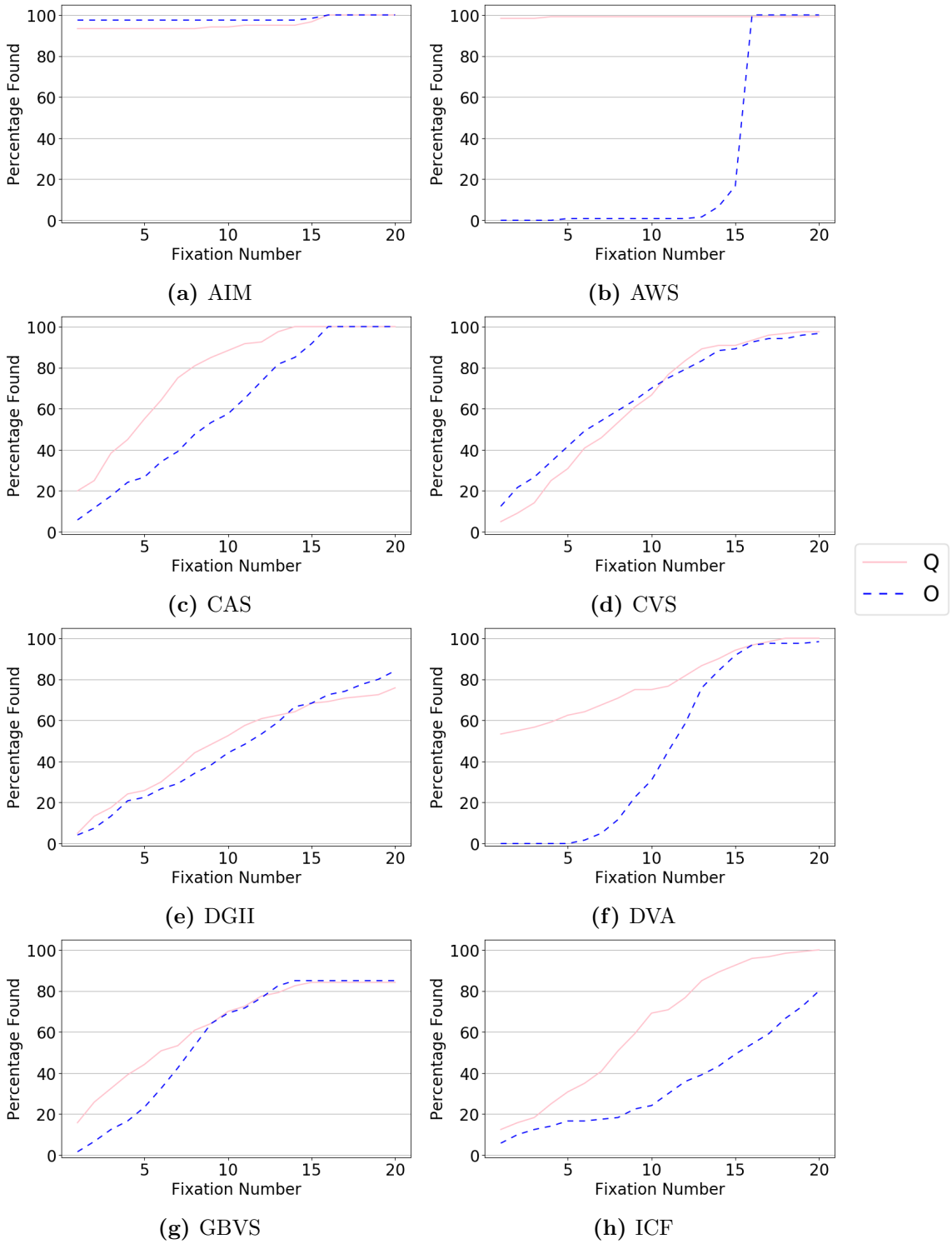
AIM found the target first in most trials regardless of whether it was a Q or an O (and was, in fact, only matched by AWS searching for a Q target), but showed a small (but significant) increase in maximum saliency ratio for the Q target than when the target was an O. AWS and DVA were

both able to find the target in almost every trial, but showed high disparity in the rapidity of finding the target. Both algorithms found a Q target quite quickly (with AWS finding it almost always on the first step), whereas there was no trial for either model in which an O target was located first.

CAS, IMSIG, and RARE2012 were also all able to successfully find the target in every trial, but only rarely did so first. ICF and oSALICON had similar performance, but exhibited a moderate (ICF) or small (oSALICON) rate of error for the O target condition. All five models also showed a moderate (CAS) to large (ICF, IMSIG, RARE2012, and oSALICON) increase in target acquisition for Q targets over O targets.

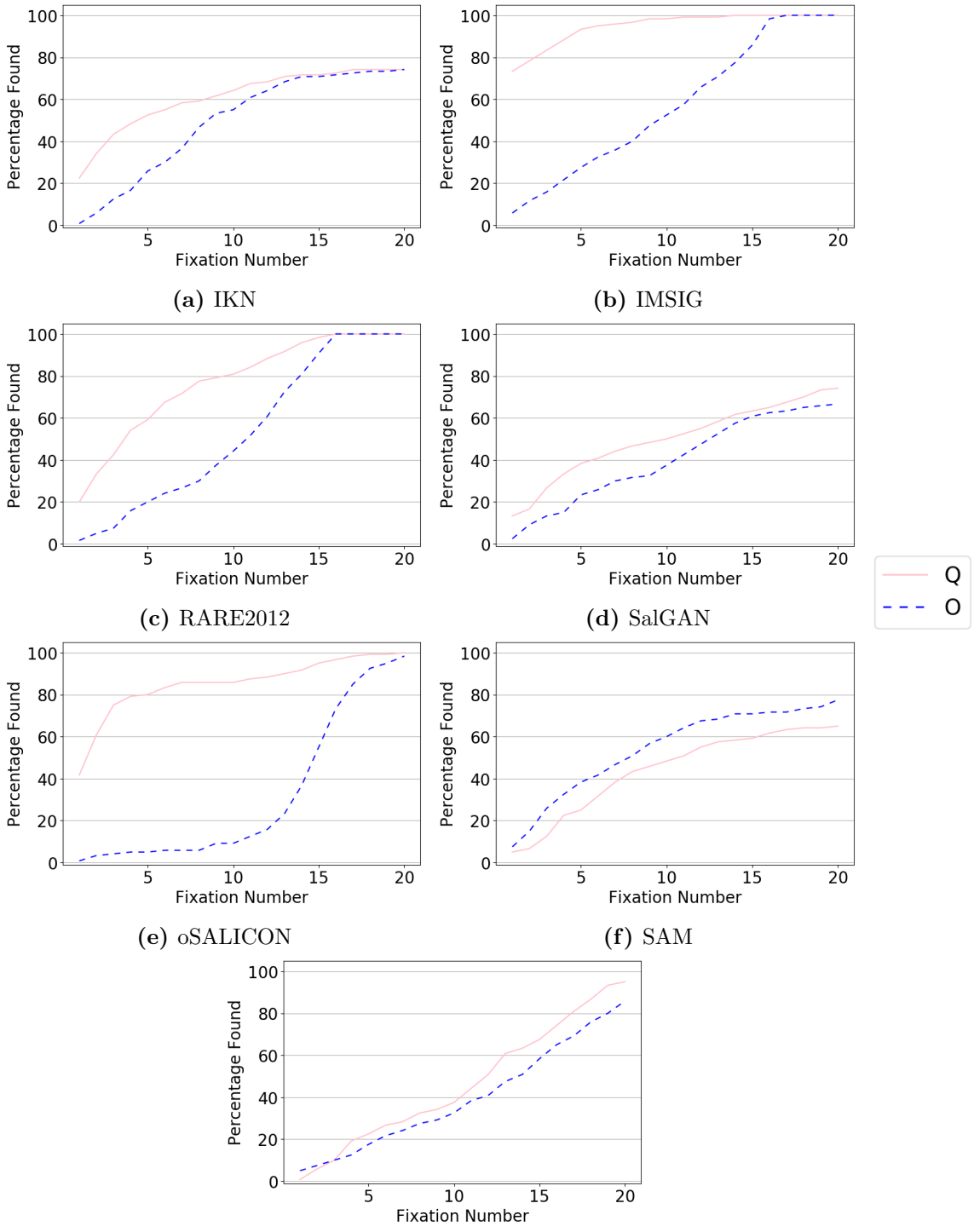
CVS could usually find the target eventually and exhibited a relatively low error rate, but showed an unusual preference for O targets over Q targets. SAM, likewise, showed a small but consistent improvement in the case of an O target over a Q target, but performed at best moderately in both cases. It is unclear why either model would show a preference for O targets, though it is interesting to see.

This asymmetry condition is clearly better represented within saliency models than the blue versus magenta condition investigated above. Nevertheless, though the vast majority of models exhibit a performance asymmetry in the expected direction, that does not mean that they fully reflect expected performance. The rapidity of target location for the AIM model is superhuman, though as mentioned in Section 4.2.2.3 this may not be due to a deviation from biological fidelity on the part of AIM, but could rather be due to the fact that there is no retinal anisotropy present in the saliency representation (see also Chapter 5 for further discussion of this topic). The need to investigate every distractor before finally locating the target in most trials (*e.g.* AWS, oSALICON) is also not very human-like, and likely represents an overly rigid internal representation of saliency that fails to adequately take into account contextual cues. Models with overly high error rates in either one (QSS, ICF) or both (DGII, IKN, GBVS, SalGAN, SSR) target conditions also deviate from expected human-like performance. The performance of CAS, IMSIG, and RARE2012 are therefore the closest to what is expected when modelling human performance in this condition. It is worth noting that oSALICON is the only model based on deep learning techniques that has an acceptably low error rate, reinforcing the observations of Section 4.1 that deep learning models of saliency struggle to extend their performance to saliency tasks in psychophysical stimuli.



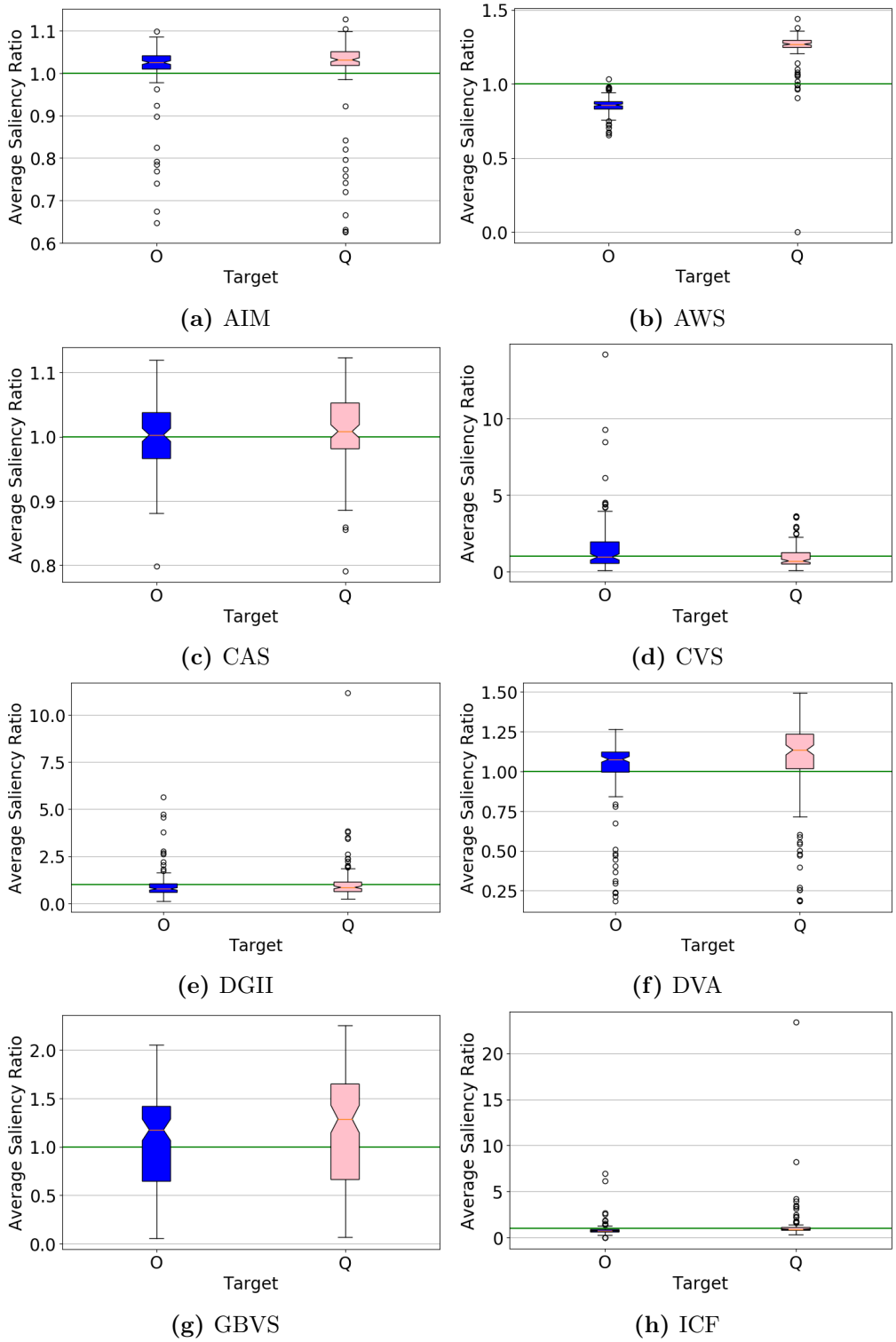
(a) AIM **(b)** AWS **(c)** CAS **(d)** CVS **(e)** DGII **(f)** DVA **(g)** GBVS **(h)** ICF

Figure 4.19: The cumulative probability of fixating the target for a given number of steps for models AIM-ICF when searching for a Q target amongst O distractors (pink line) or when searching for an O target amongst Q distractors (dashed blue line).



(a) IKN **(b)** IMSIG **(c)** RARE2012 **(d)** SalGAN **(e)** oSALICON **(f)** SAM **(g)** SSR

Figure 4.20: The cumulative probability of fixating the target for a given number of steps for models IKN-SSR when searching for a Q target amongst O distractors (pink line) or when searching for an O target amongst Q distractors (dashed blue line).



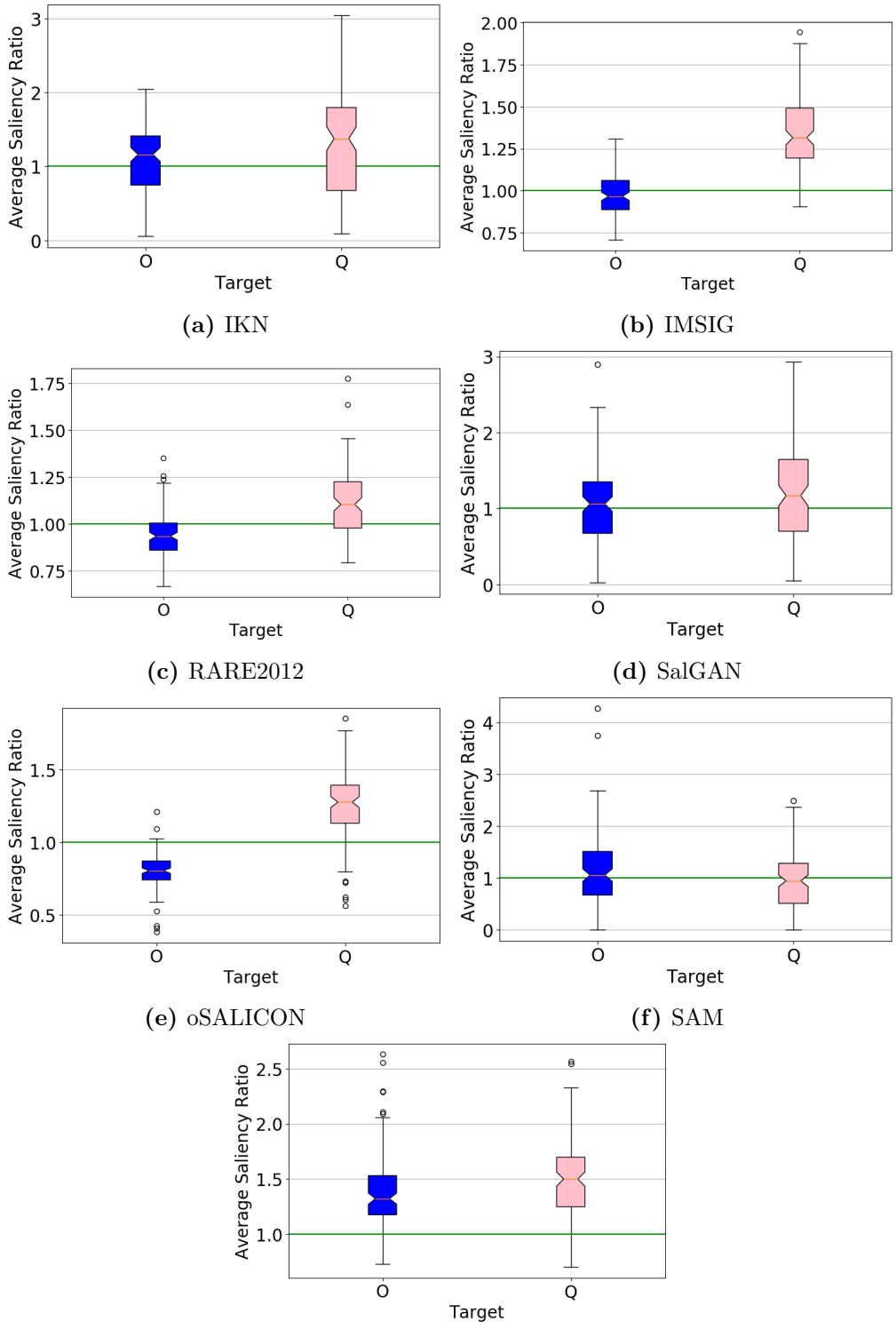
(a) AIM **(b)** AWS

(c) CAS **(d)** CVS

(e) DGII **(f)** DVA

(g) GBVS **(h)** ICF

Figure 4.21: The ratio of target to distractor average saliency values for AIM through ICF. Results on the left in each plot (blue) are for O targets amongst Q distractors, whereas results on the right (pink) are for Q targets amongst O distractors.



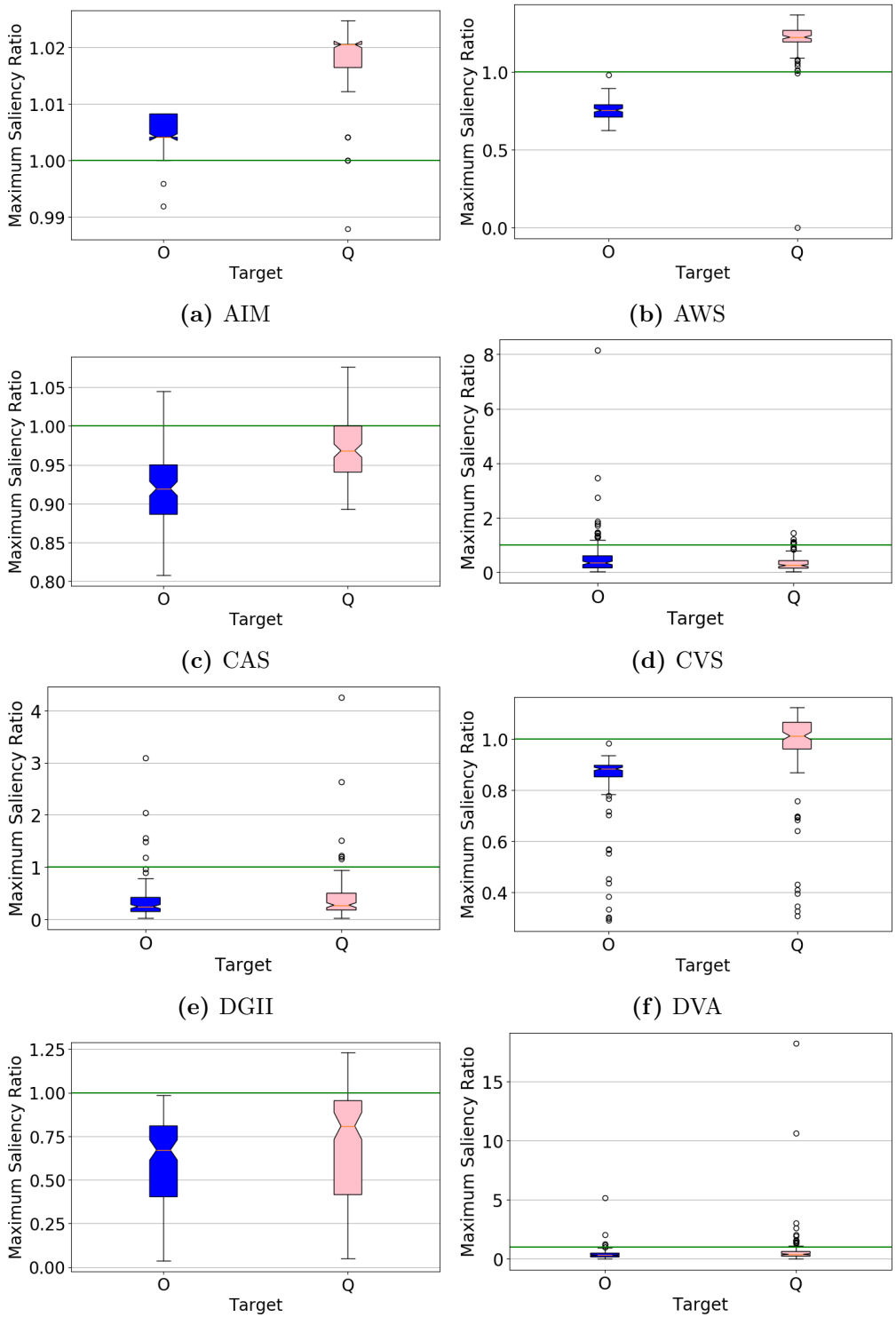
(a) IKN **(b)** IMSIG

(c) RARE2012 **(d)** SalGAN

(e) oSALICON **(f)** SAM

(g) SSR

Figure 4.22: The ratio of target to distractor average saliency values for IKN through SSR. Results on the left in each plot (blue) are for O targets amongst Q distractors, whereas results on the right (pink) are for Q targets amongst O distractors.



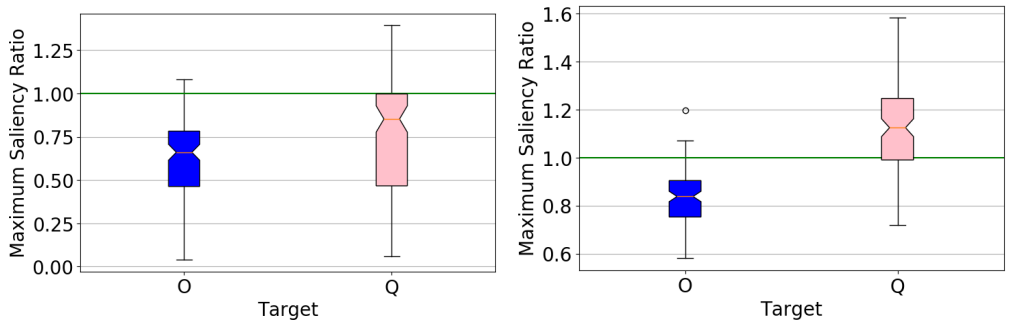
(a) AIM **(b)** AWS

(c) CAS **(d)** CVS

(e) DGII **(f)** DVA

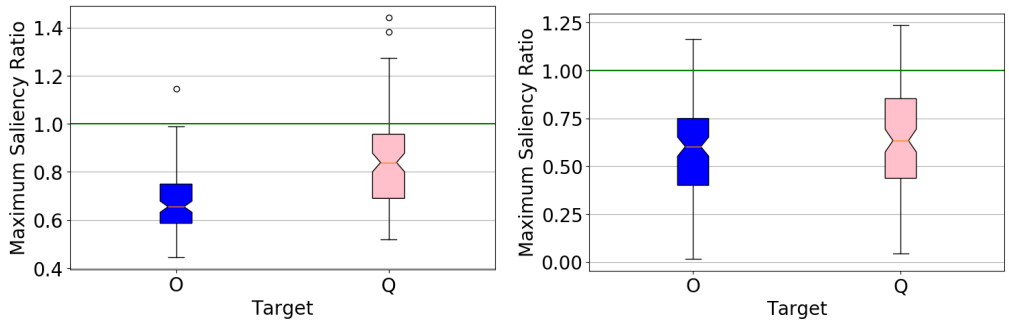
(g) GBVS **(h)** ICF

Figure 4.23: The ratio of target to distractor maximum saliency values for AIM through ICF. Results on the left in each plot (blue) are for O targets amongst Q distractors, whereas results on the right (pink) are for Q targets amongst O distractors.



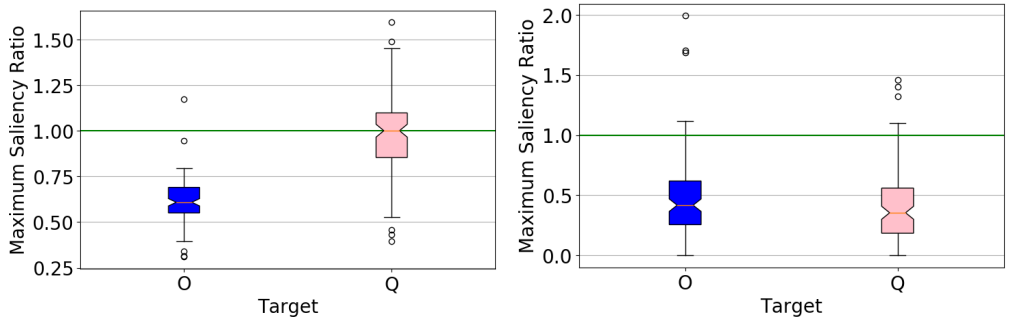
(a) IKN

(b) IMSIG



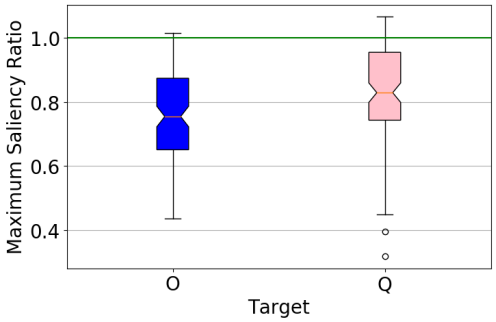
(c) RARE2012

(d) SalGAN



(e) oSALICON

(f) SAM



(g) SSR

Figure 4.24: The ratio of target to distractor maximum saliency values for IKN through SSR. Results on the left in each plot (blue) are for O targets amongst Q distractors, whereas results on the right (pink) are for Q targets amongst O distractors.

4.2.3.1.3 Flipped A vs. Canonical A: The results for visual search arrays formed by flipped and canonically oriented A's (example shown in Figure 4.8) are presented in Figures 4.25-4.30. Figures 4.25 and 4.26 show the cumulative probability of fixating the target as the number of steps increases. Figures 4.27 and 4.28 show the average saliency ratio, while the maximum saliency ratio is shown in Figures 4.29 and 4.30.

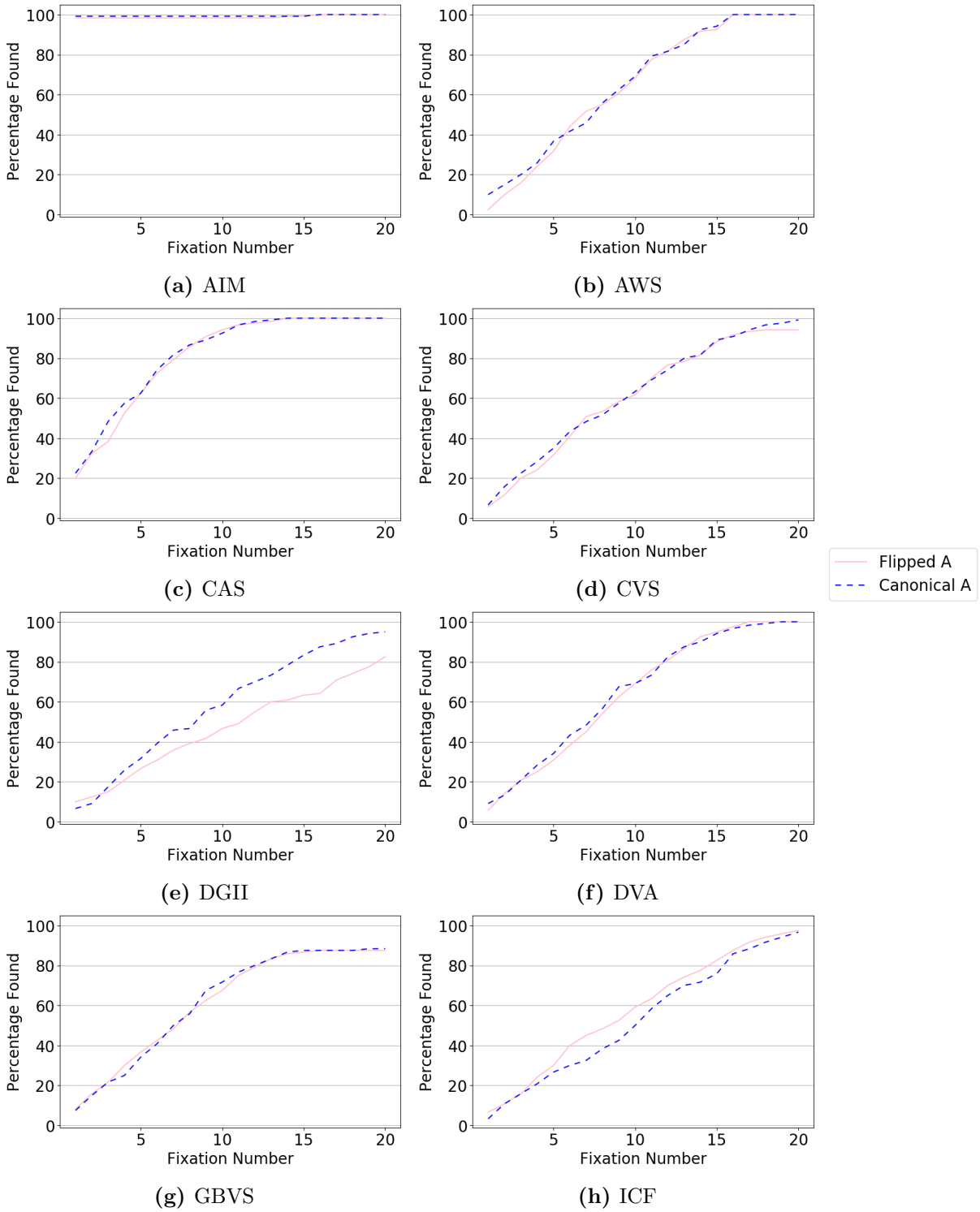
Although letter arrays with inverted or reversed elements form a well-established class of search asymmetry, they rely on a degree of familiarity with the canonical appearance of the element. This asymmetry condition therefore represents the first "novelty" asymmetry condition. Although it is reasonable to expect classic saliency models to locate the singleton target, there is no particular reason to expect that they will exhibit the expected asymmetry, and therefore the discussion of performance here and for subsequent asymmetry conditions will focus on those models that are based on learning (including ICF; though its features are explicitly not based on deep learning, the readout network that interprets them is trained according to deep learning techniques). The results of all models are nevertheless provided for completeness.

As with the O and Q condition, oSALICON is the only deep learning model to exhibit a zero error rate, successfully managing to find the target in both the flipped and not flipped condition. However, it shows no clear asymmetry in performance; the average saliency ratio of canonical A targets is slightly higher than for flipped A targets, whereas the maximum saliency ratio exhibits the opposite pattern (in neither case are the ratios significantly different, and the average target saliency ratio in both conditions hovers near one). oSALICON therefore appears to be assigning nearly identical values of salience to all scene elements, and then traverses the image in what is essentially a random order to look for the target. This invariably finds the target, but fails to adequately account for human behaviour. ICF appears to exhibit similar behaviour, though with a small error frequency in which the target is never found.

SalGAN shows a lack of performance asymmetry and fails to locate the target with adequate accuracy in either condition. DGII does show a performance asymmetry, though the pattern is rather unusual. In terms of target acquisition and error rate, it clearly favours the canonical A target, but it nevertheless shows a significantly higher maximum saliency ratio for flipped targets (though still below 1 on average). It achieves a relatively low rate of error (3.3%) in the canonically oriented A target condition, and a much higher error rate (14.2%) in the flipped target condition.

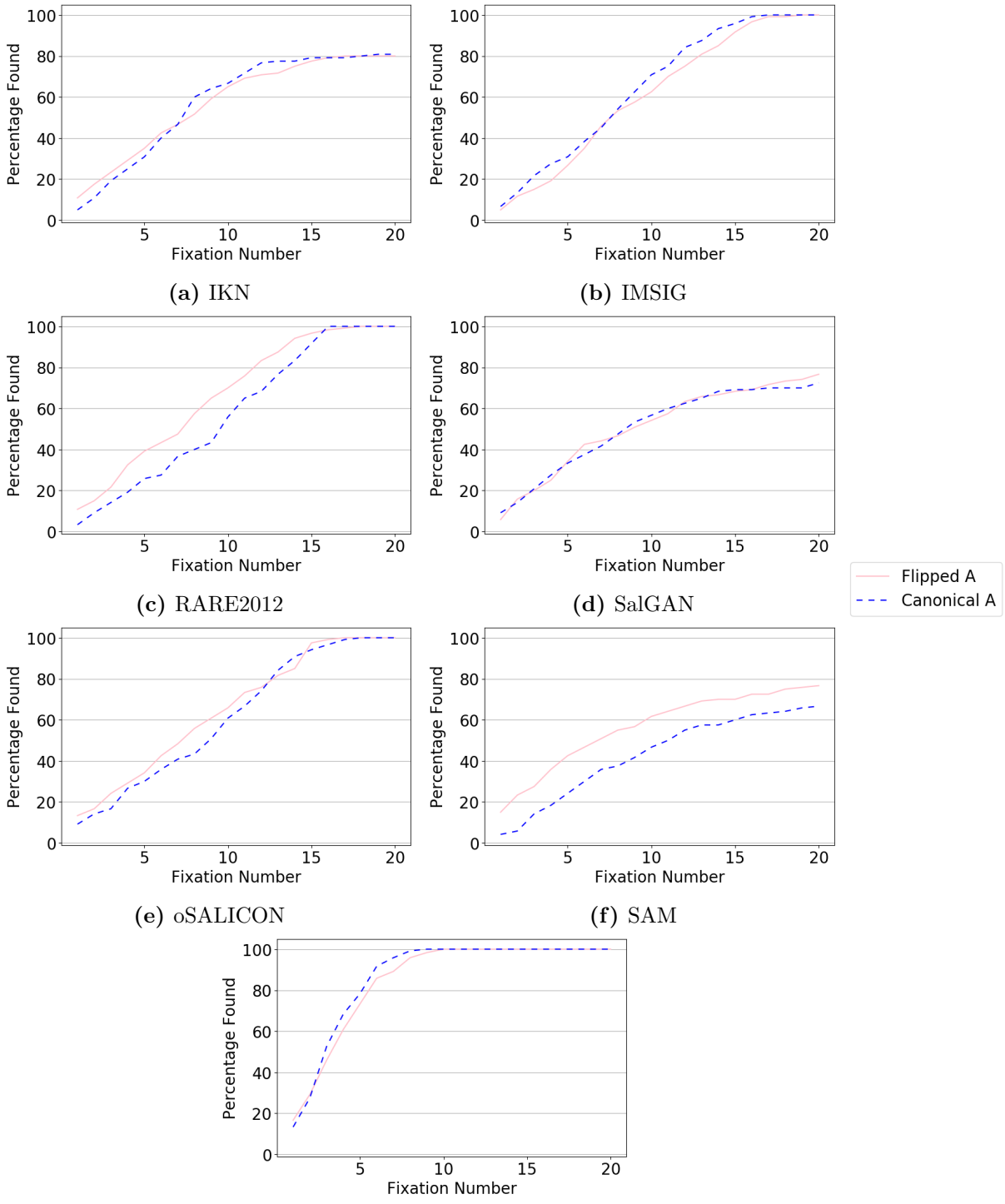
SAM does show the expected performance asymmetry, detecting flipped A targets more quickly (and with fewer errors) than canonically oriented targets and assigning significantly higher saliency to flipped targets, but the error rates in both target conditions remain unacceptably high.

Deep learning based models not only do not appear to consistently reflect the same kind of performance asymmetry that humans exhibit on this task, they also appear to struggle with the task itself. This task does appear to be a more difficult search task for classic models, too, with many of them showing performance that appears to assign an approximately equal degree of salience to all scene elements (much like oSALICON and ICF). A few models appear to achieve high performance (AIM finds the target first in all trials and shows a significant, albeit small, boost in maximum saliency ratio for flipped targets, while CAS and SSR consistently find the target more rapidly than a chance traversal of elements), though most models show little if any significant asymmetry (as expected, given their lack of learning).



(a) AIM **(b)** AWS **(c)** CAS **(d)** CVS **(e)** DGII **(f)** DVA **(g)** GBVS **(h)** ICF

Figure 4.25: The cumulative probability of fixating the target for a given number of steps for models AIM-ICF when searching for a flipped A target amongst canonical A distractors (pink line) or when searching for a canonical A target amongst flipped A distractors (dashed blue line).



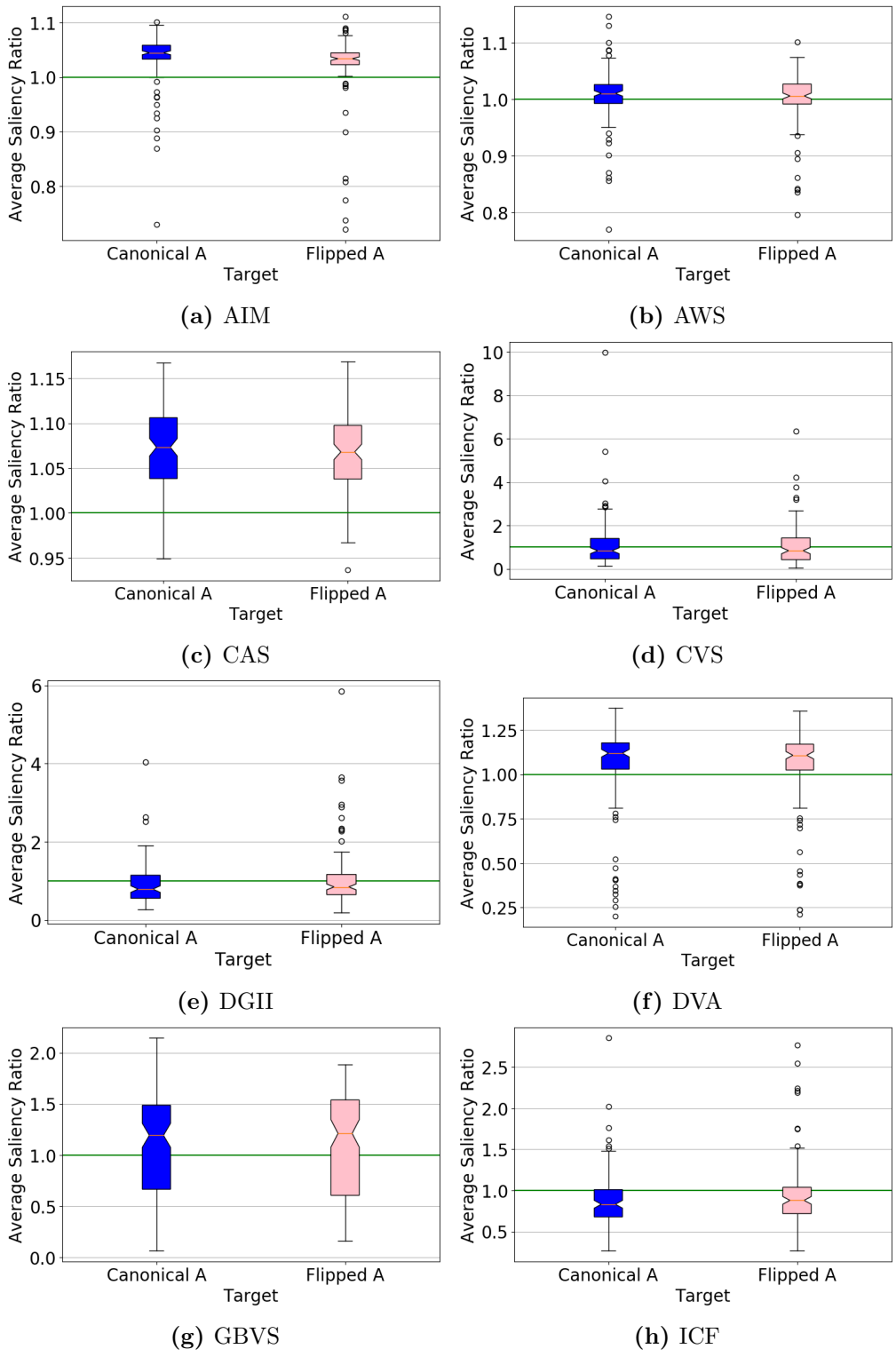
(a) IKN **(b)** IMSIG

(c) RARE2012 **(d)** SalGAN

(e) oSALICON **(f)** SAM

(g) SSR

Figure 4.26: The cumulative probability of fixating the target for a given number of steps for models IKN-SSR when searching for a flipped A target amongst canonical A distractors (pink line) or when searching for a canonical A target amongst flipped A distractors (dashed blue line).



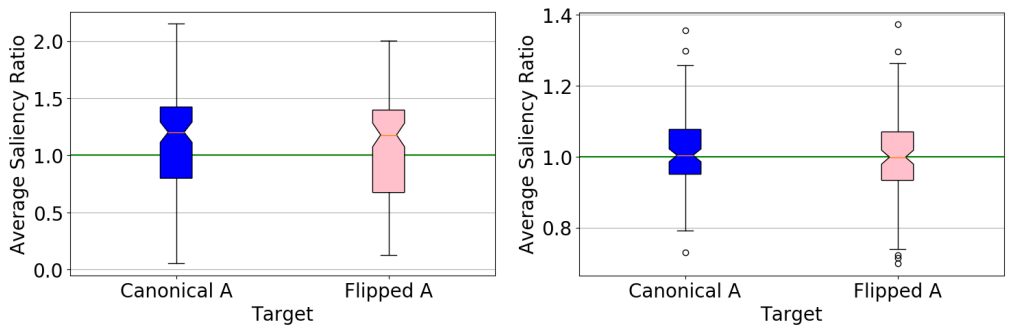
(a) AIM **(b)** AWS

(c) CAS **(d)** CVS

(e) DGII **(f)** DVA

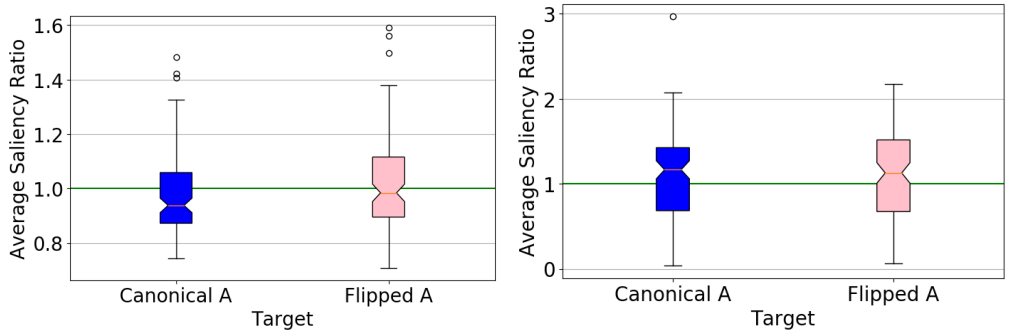
(g) GBVS **(h)** ICF

Figure 4.27: The ratio of target to distractor average saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical A targets amongst flipped A distractors, whereas results on the right (pink) are for flipped A targets amongst canonical A distractors.



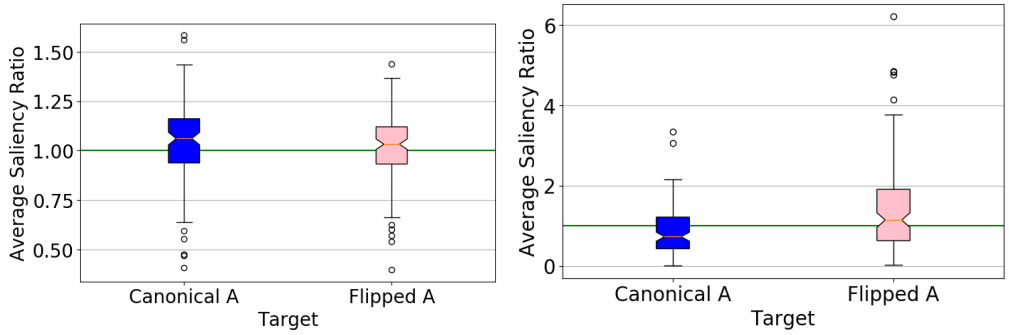
(a) IKN

(b) IMSIG



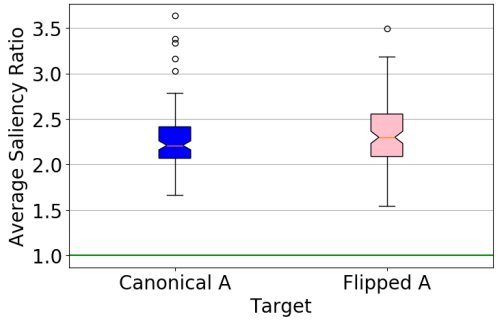
(c) RARE2012

(d) SalGAN



(e) oSALICON

(f) SAM



(g) SSR

Figure 4.28: The ratio of target to distractor average saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical A targets amongst flipped A distractors, whereas results on the right (pink) are for flipped A targets amongst canonical A distractors.

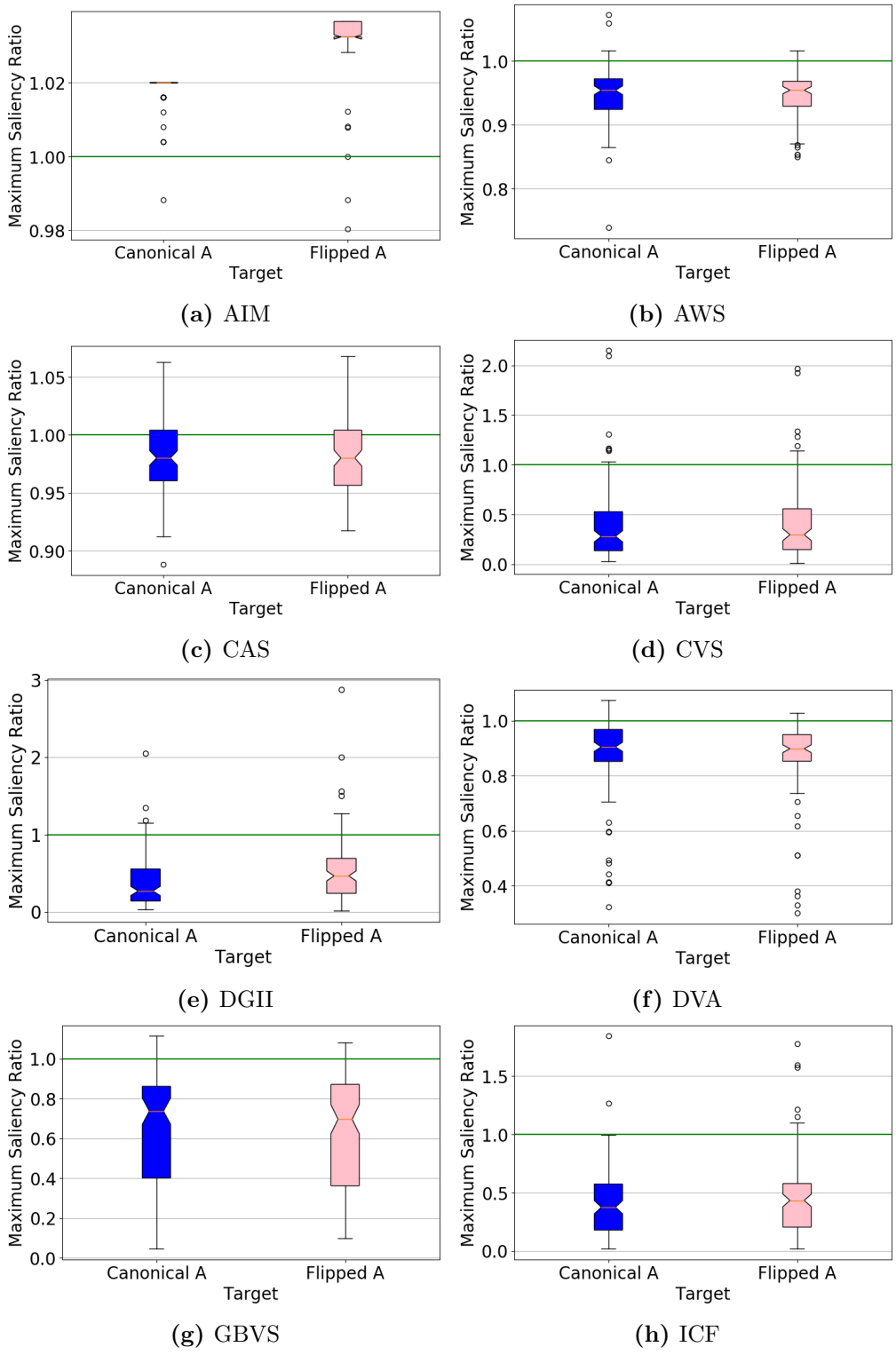
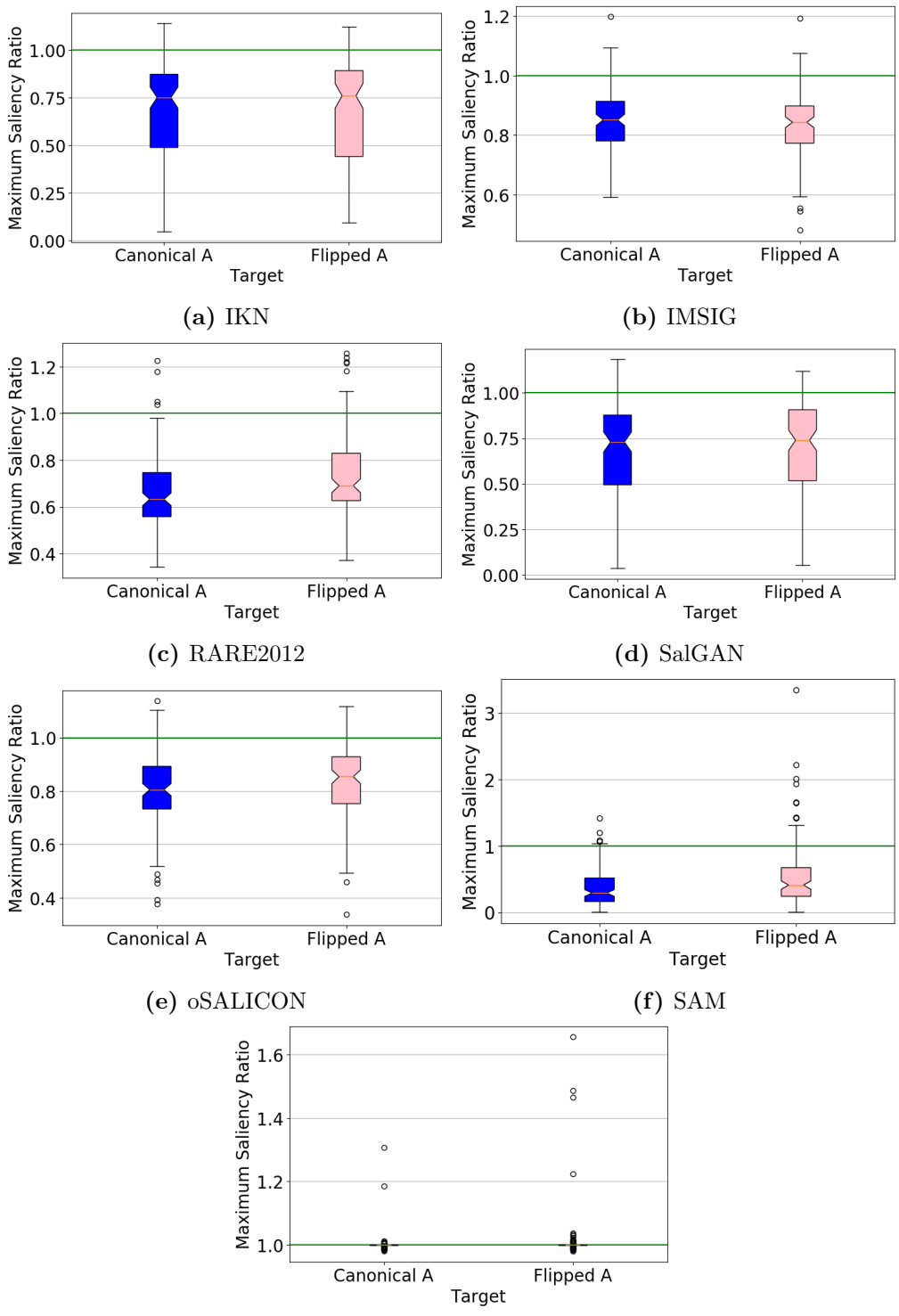


Figure 4.29: The ratio of target to distractor maximum saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical A targets amongst flipped A distractors, whereas results on the right (pink) are for flipped A targets amongst canonical A distractors.



(a) IKN **(b)** IMSIG

(c) RARE2012 **(d)** SalGAN

(e) oSALICON **(f)** SAM

(g) SSR

Figure 4.30: The ratio of target to distractor maximum saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical A targets amongst flipped A distractors, whereas results on the right (pink) are for flipped A targets amongst canonical A distractors.

4.2.3.1.4 Flipped Q vs. Canonical Q: The results for visual search arrays formed by flipped and canonically oriented Q's (example shown in Figure 4.9) are presented in Figures 4.31-4.36. Figures 4.31 and 4.32 show the cumulative probability of fixating the target as the number of steps increases. Figures 4.33 and 4.34 show the average saliency ratio, while the maximum saliency ratio is shown in Figures 4.35 and 4.36.

Similar to the canonical and flipped A condition, this experimental condition is included as an example of a search asymmetry based on letter familiarity. However, whereas A's represent rather common letters and might be expected to be well represented in a dataset including text, Q's are generally rarer in written English and this condition therefore provides a counterpoint to the results found in the experiment with A's.

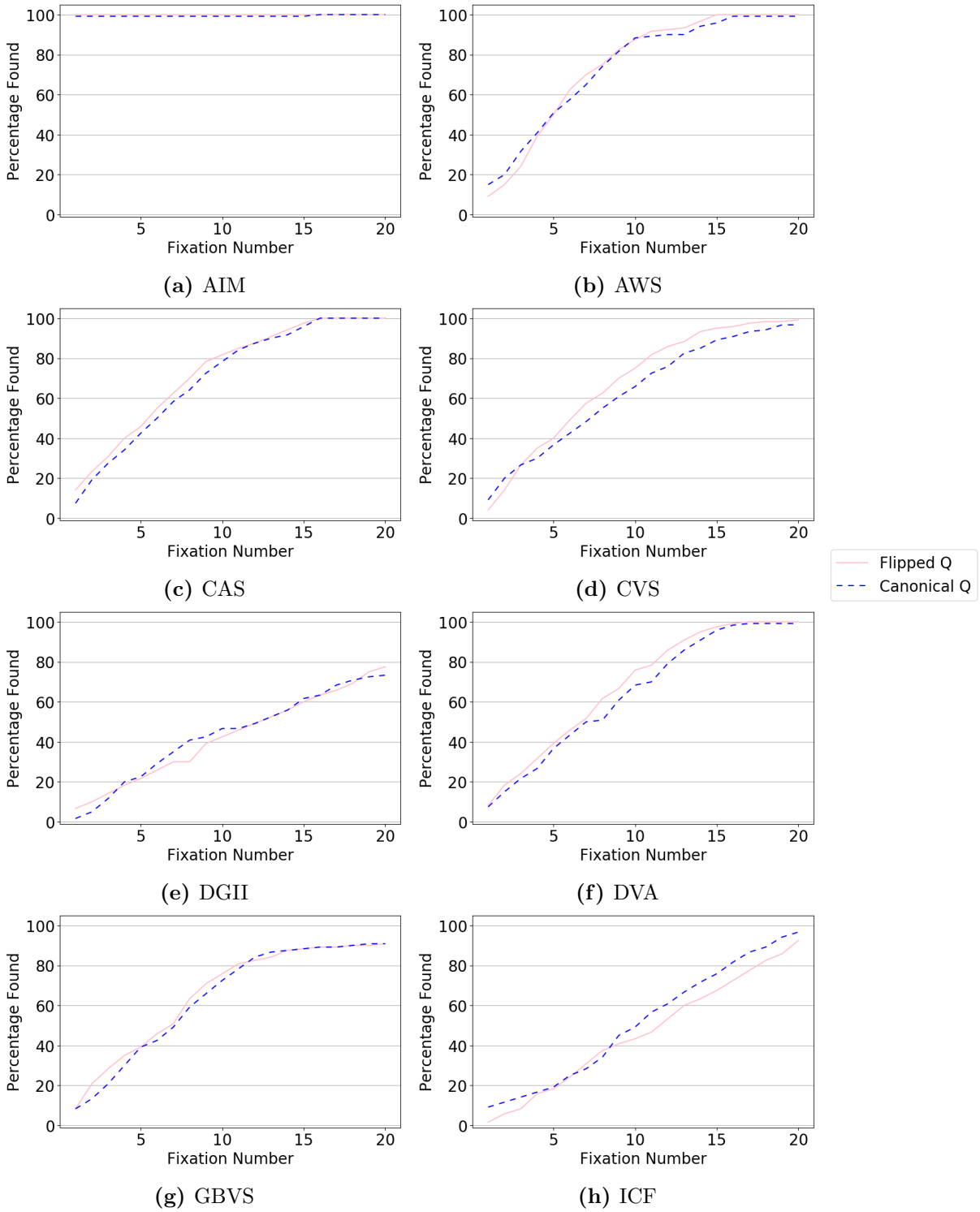
DGII and SAM exhibit relatively poor performance and no significant asymmetry between flipped and canonical conditions. ICF also has no significant asymmetry, but while it does not find the target quickly it does achieve a much lower error rate than DGII and SAM. SalGAN exhibits a degree of preference for the canonically oriented Q target, both achieving a lower error rate and faster average detection speed for canonical Q's compared to flipped Q's. The overall performance of SalGAN, however, remains close to the rather poor behaviour of DGII and SAM.

oSALICON again shows the best overall performance of the deep learning models tested in terms of error rate, and unlike with the flipped and canonical A experiment, here oSALICON exhibits an extremely strong performance asymmetry and significant difference in saliency ratios. However, the asymmetry is in the opposite direction expected, with the vast majority of flipped Q targets being found only after the majority of canonically oriented distractors have already been investigated. It is unclear why oSALICON would show such a strong asymmetry in this case but not for the letter A.

As with the previous flipped letter experiment, this appears to have been challenging not only for the deep learning models but also for most of the classic models, with the exception of AIM (which again was able to find the target consistently on the first step for both target conditions). A number of additional models (AWS, CAS, CVS, DVA, IMSIG and RARE2012) were able to achieve very low or zero rates of error, but the speed of location for most of these models is not much faster than a random traversal of array elements. Aside from AIM and IMSIG, which both showed a small but significant preference for flipped Q targets in terms of maximum saliency ratios, none of the

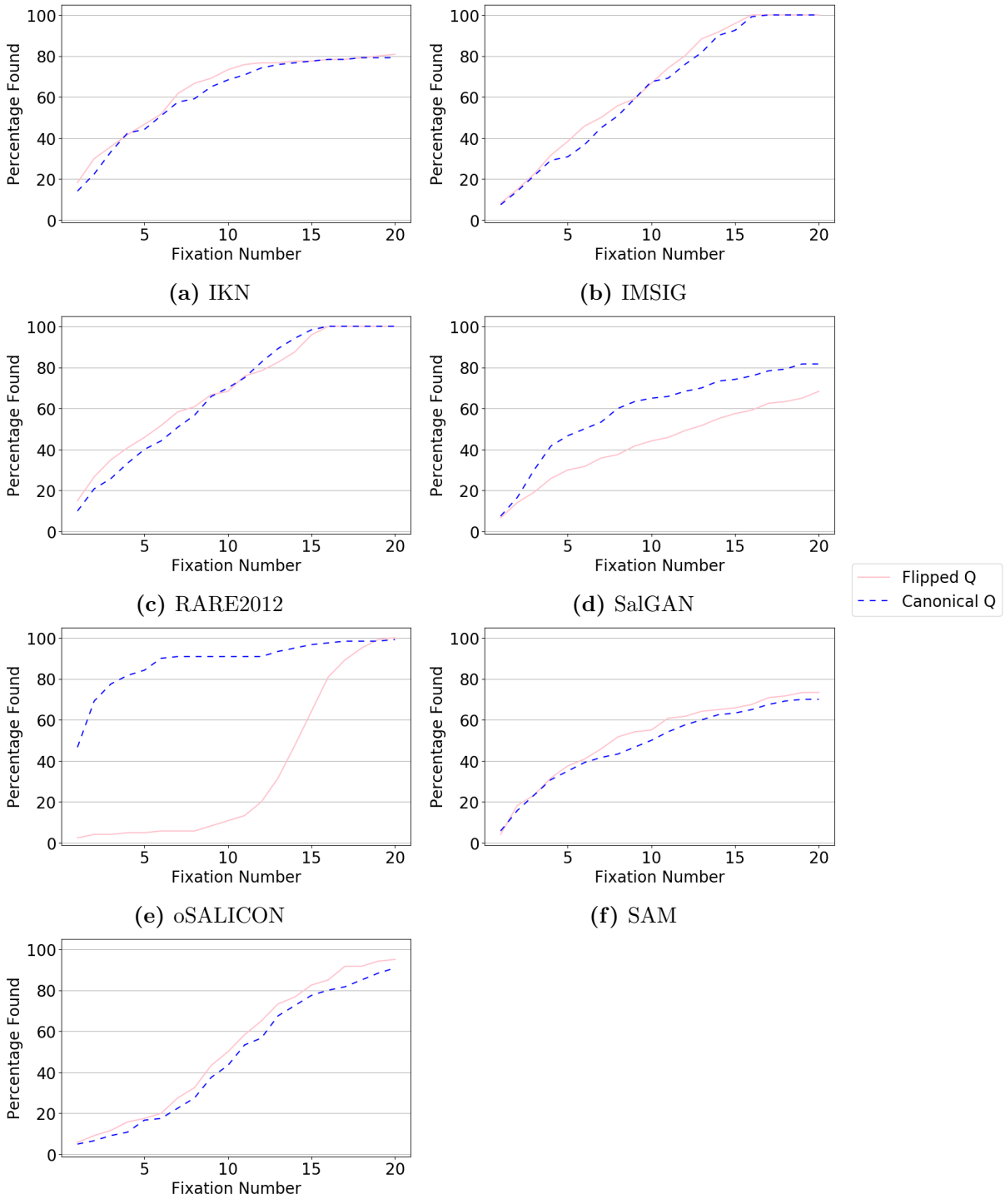
classical models show a significant degree of asymmetry. In terms of acquisition performance, CVS and DVA show the largest effect with a minor difference in error and acquisition rates favouring the flipped targets.

Overall for both flipped and canonical letter experiments, oSALICON is the only deep learning model to achieve a low miss rate, and none of the models, with the exception of SAM and DGII operating over flipped and canonical A's, significantly exhibit the expected asymmetry. Although text is certainly a common element in natural images (for example, through signage), it is possible that there is simply not enough emphasis on text in the training data used for the deep models tested here to achieve the expected performance. The next series of experiments, therefore, will test more complex objects for which many of the features used by these models will have been explicitly tuned towards.



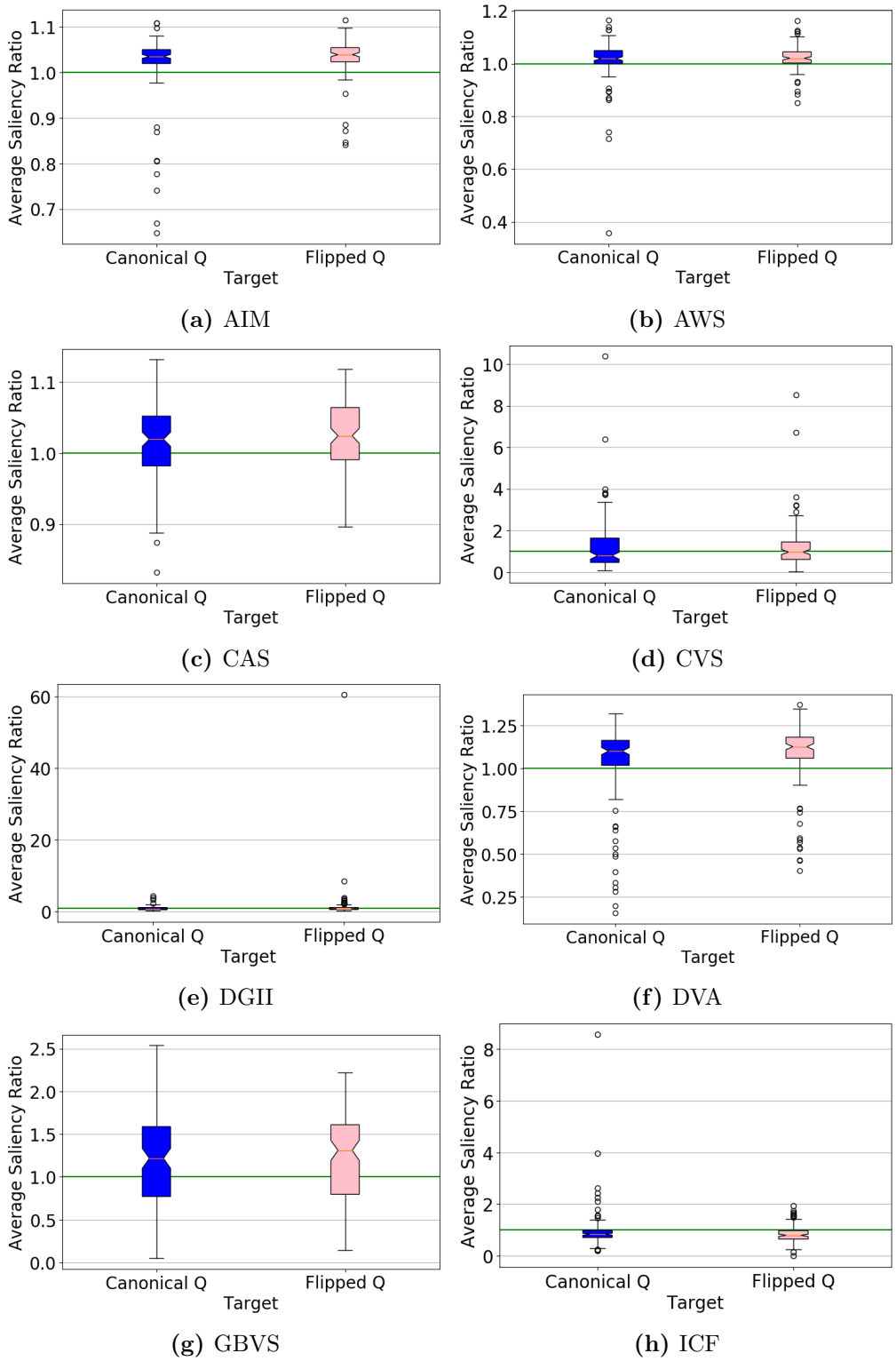
(a) AIM **(b)** AWS **(c)** CAS **(d)** CVS **(e)** DGII **(f)** DVA **(g)** GBVS **(h)** ICF

Figure 4.31: The cumulative probability of fixating the target for a given number of steps for models AIM-ICF when searching for a flipped Q target amongst canonical Q distractors (pink line) or when searching for a canonical Q target amongst flipped Q distractors (dashed blue line).



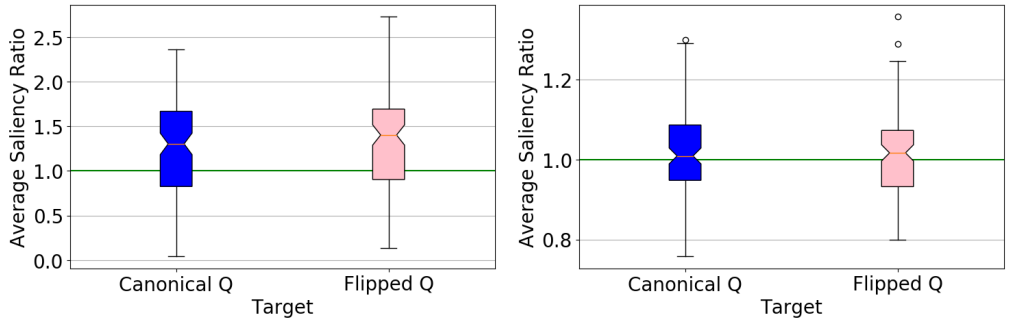
(a) IKN **(b)** IMSIG **(c)** RARE2012 **(d)** SalGAN **(e)** oSALICON **(f)** SAM **(g)** SSR

Figure 4.32: The cumulative probability of fixating the target for a given number of steps for models IKN-SSR when searching for a flipped Q target amongst canonical Q distractors (pink line) or when searching for a canonical Q target amongst flipped Q distractors (dashed blue line).



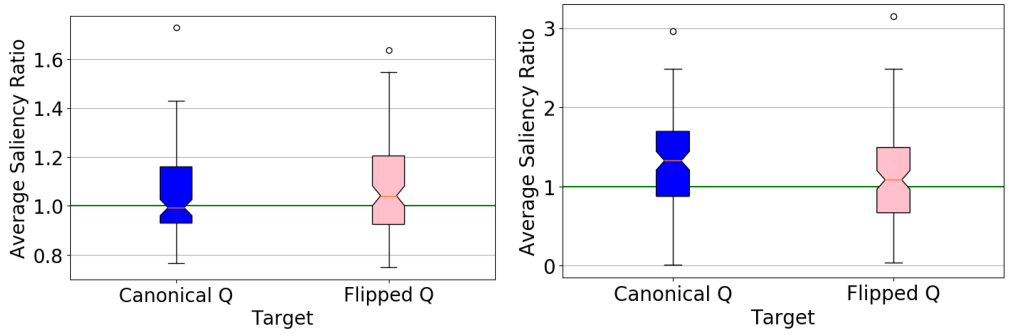
(a) AIM **(b)** AWS **(c)** CAS **(d)** CVS **(e)** DGII **(f)** DVA **(g)** GBVS **(h)** ICF

Figure 4.33: The ratio of target to distractor average saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical Q targets amongst flipped Q distractors, whereas results on the right (pink) are for flipped Q targets amongst canonical Q distractors.



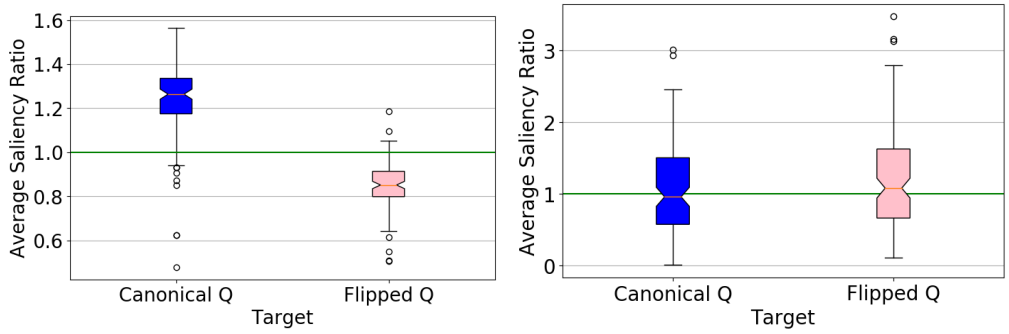
(a) IKN

(b) IMSIG



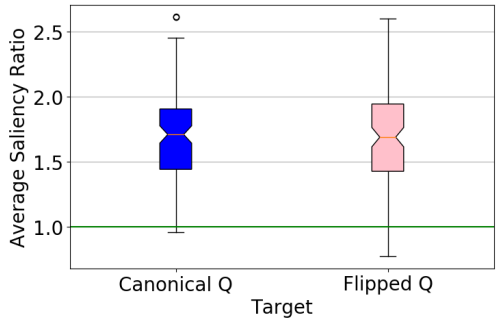
(c) RARE2012

(d) SalGAN



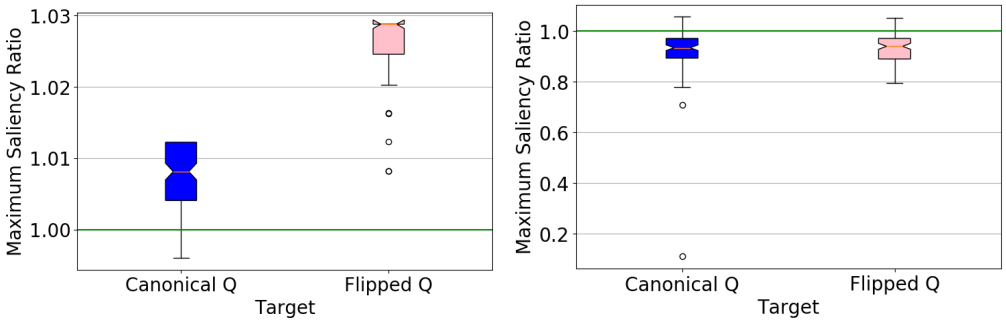
(e) oSALICON

(f) SAM



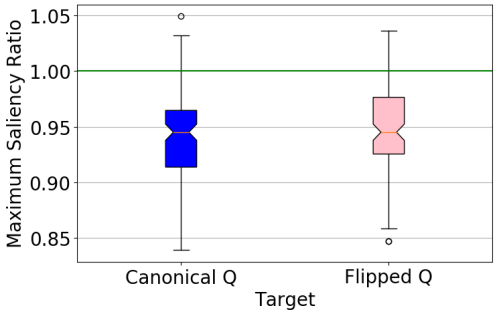
(g) SSR

Figure 4.34: The ratio of target to distractor average saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical Q targets amongst flipped Q distractors, whereas results on the right (pink) are for flipped Q targets amongst canonical Q distractors.

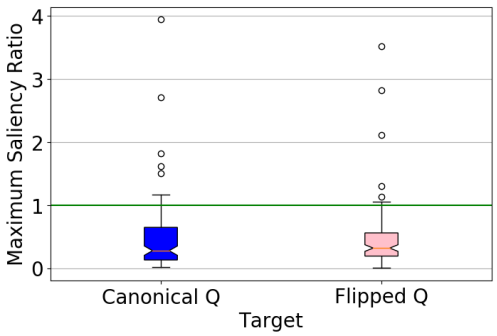


(a) AIM

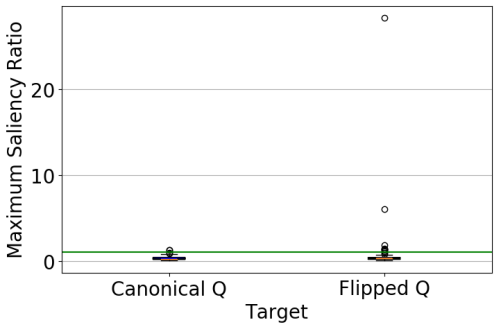
(b) AWS



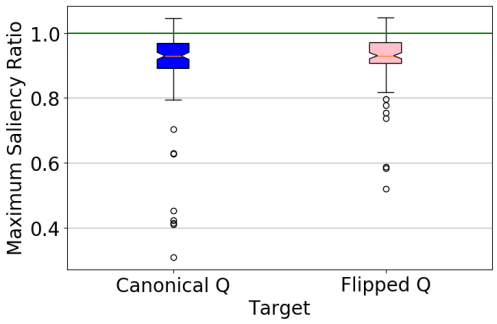
(c) CAS



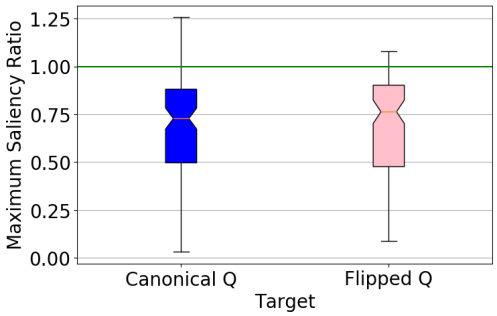
(d) CVS



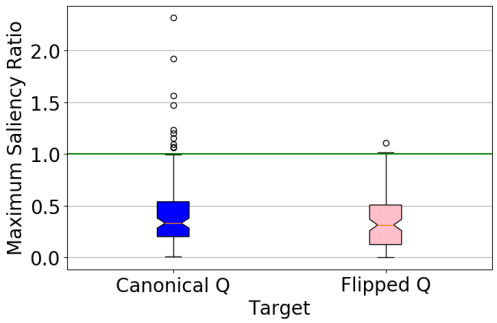
(e) DGII



(f) DVA

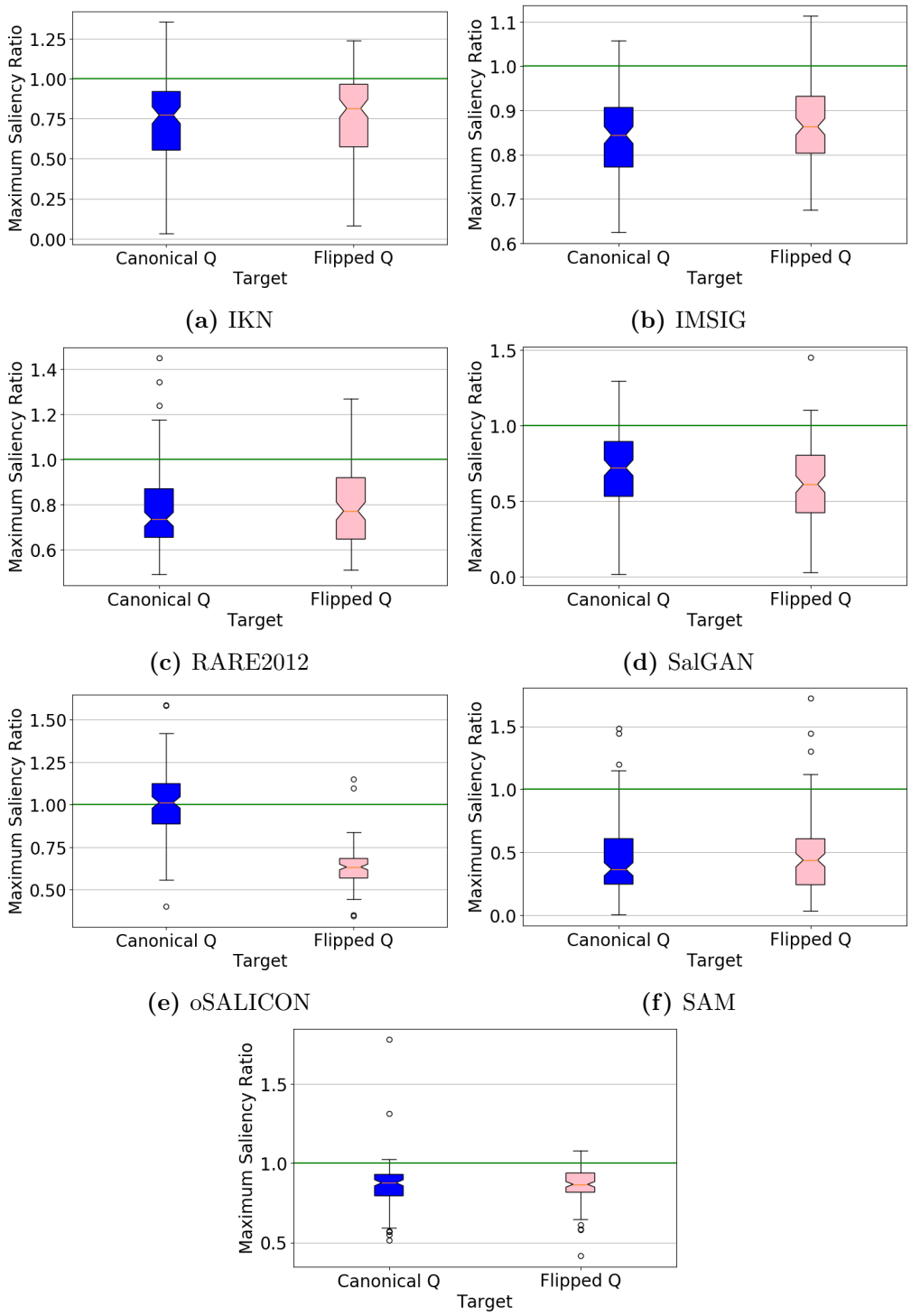


(g) GBVS



(h) ICF

Figure 4.35: The ratio of target to distractor maximum saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical Q targets amongst flipped Q distractors, whereas results on the right (pink) are for flipped Q targets amongst canonical Q distractors.



(a) IKN **(b)** IMSIG

(c) RARE2012 **(d)** SalGAN

(e) oSALICON **(f)** SAM

(g) SSR

Figure 4.36: The ratio of target to distractor maximum saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical Q targets amongst flipped Q distractors, whereas results on the right (pink) are for flipped Q targets amongst canonical Q distractors.

4.2.3.2 Complex Asymmetries

This section presents results from asymmetry tests over scene elements for which the constituent deep features have been trained in the deep learning models of saliency. Table 4.5 presents the percentage of trials for which targets were not found after 21 steps, while Tables 4.6 and 4.7 present the average differences in average and maximum saliency ratios, respectively, between the paired asymmetric target conditions. Specific patterns of results are discussed below. As with the letter-based asymmetries above, most discussion will focus on the performance of deep learning based models. Classical model performance is included for completeness, and will be noted when of interest.

Overall it should be noted that performance for the deep learning models is improved in this set of experiments compared to the more classic stimuli tested previously (in which oSALICON was the only model to achieve an error rate of at most 5%, with SalGAN and SAM achieving relatively high error in all conditions and DGII and ICF suffering from more sporadic spikes in miss rate). As seen in Table 4.5, DGII, ICF, and oSALICON all manage to find the target in every trial, and SalGAN and SAM achieve less than 10% error in all but one condition, and even then it is very close to 10%.

It is also worth noting that this improvement is not unique to the deep learning models; even more models which include errors in all target conditions (CVS, GBVS, IKN, RARE2012, and SSR), even the highest error rates (SSR) do not exceed 10%. It does make sense that performance would be improved in this set of experiments; although the search arrays are clearly not natural images, the use of complex and detailed objects as target and distractor elements provides features that are closer to those which all the algorithms tested here have primarily been tested on previously. Even the mask images are typically at least as feature-rich as letter arrays, and are more likely to contain a distribution of features more similar to those seen in natural images.

4.2.3.2.1 Fully Detailed Elements: The results for visual search arrays formed by flipped and canonically oriented complex elements (using the person elements shown in Figures 4.10 and arranged in random search arrays as in Figure 4.11) are presented in Figures 4.37-4.42. Figures 4.37 and 4.38 show the cumulative probability of fixating the target as the number of steps increases. Figures 4.39 and 4.40 show the average saliency ratio, while the maximum saliency ratio is shown

Target	AIM	AWS	CAS	CVS	DGII	DVA	GBVS	ICF
Person	0	0	0	2.4	0	0	2.9	0
Flipped Person	0	0	0	2.4	1.9	0.5	5.2	1.4
Person Mask	0.7	0	0	5.7	0.4	0.4	5.0	0.7
Flipped Person Mask	0	0.7	0	4.6	9.3	0.7	5.7	1.1
Target	**IKN**	**IMSIG**	**QSS**	**RARE2012**	**SalGAN**	**oSALICON**	**SAM**	**SSR**
Person	5.2	1.9	0	1.0	11.4	5.7	17.1	6.2
Flipped Person	4.8	4.8	0	0.5	23.8	7.6	23.8	6.2
Person Mask	5.4	3.6	-	3.9	12.9	9.6	16.8	9.6
Flipped Person Mask	5.4	6.8	-	4.6	23.9	10.4	21.4	6.1

Table 4.5: Table showing the percentage of missed targets for each target-distractor condition for the complex asymmetries conditions. Note that for each target the distractors were vertically inverted copies of the target. QSS did not provide a valid output map in the images based on binary masks, and was therefore excluded from that condition.

Target A	Target B	AIM	AWS	CAS	CVS
Person	Flipped Person	0.001 (0.49)	0.002 (0.68)	-0.003 (0.74)	0.114 (0.18)
Person Mask	Flipped Person Mask	*0.002 (0.02)*	-0.001 (0.80)	0.003 (0.50)	-0.035 (0.91)
Target A	**Target B**	**DGII**	**DVA**	**GBVS**	**ICF**
Person	Flipped Person	***-0.302 (0.00)***	0.002 (0.83)	-0.030 (0.50)	-0.004 (0.78)
Person Mask	Flipped Person Mask	0.003 (0.91)	0.001 (0.95)	0.025 (0.56)	0.018 (0.47)
Target A	**Target B**	**IKN**	**IMSIG**	**QSS**	**RARE2012**
Person	Flipped Person	-0.025 (0.35)	***0.076 (0.00)***	0.005 (0.87)	-0.007 (0.61)
Person Mask	Flipped Person Mask	0.001 (0.96)	***-0.035 (0.00)***	-	*0.055 (0.02)*
Target A	**Target B**	**SalGAN**	**oSALICON**	**SAM**	**SSR**
Person	Flipped Person	***-0.195 (0.00)***	***-0.749 (0.00)***	***-0.981 (0.00)***	0.036 (0.58)
Person Mask	Flipped Person Mask	***-0.201 (0.00)***	***-0.464 (0.00)***	***-0.567 (0.00)***	-0.012 (0.76)

Table 4.6: Table showing the change in average saliency values assigned to Target B in comparison to Target A (p-value shown in parentheses). Results with $p < 0.05$ are italicized, and results with $p < 0.01$ are written in bold. Note that QSS is missing results for the mask conditions due to invalid output for these images.

Target A	**Target B**	**AIM**	**AWS**	**CAS**	**CVS**
Person	Flipped Person	0.000 (0.63)	0.004 (0.51)	-0.004 (0.45)	*0.090 (0.04)*
Person Mask	Flipped Person Mask	***0.004 (0.00)***	-0.002 (0.74)	0.002 (0.43)	0.001 (0.99)
Target A	**Target B**	**DGII**	**DVA**	**GBVS**	**ICF**
Person	Flipped Person	***-0.924 (0.00)***	0.005 (0.46)	-0.008 (0.77)	0.021 (0.63)
Person Mask	Flipped Person Mask	***-0.102 (0.00)***	*-0.015 (0.02)*	0.029 (0.25)	0.064 (0.23)
Target A	**Target B**	**IKN**	**IMSIG**	**QSS**	**RARE2012**
Person	Flipped Person	-0.016 (0.43)	0.004 (0.68)	0.004 (0.87)	0.022 (0.15)
Person Mask	Flipped Person Mask	-0.013 (0.47)	-0.003 (0.69)	-	0.038 (0.07)
Target A	**Target B**	**SalGAN**	**oSALICON**	**SAM**	**SSR**
Person	Flipped Person	***-0.113 (0.00)***	***-0.508 (0.00)***	***-0.492 (0.00)***	-0.008 (0.53)
Person Mask	Flipped Person Mask	***-0.114 (0.00)***	***-0.310 (0.00)***	***-0.279 (0.00)***	0.000 (0.73)

Table 4.7: Table showing the change in maximum saliency values assigned to Target B in comparison to Target A (p-value shown in parentheses). Results with $p < 0.05$ are italicized, and results with $p < 0.01$ are written in bold. Note that QSS is missing results for the mask conditions due to invalid output for these images.

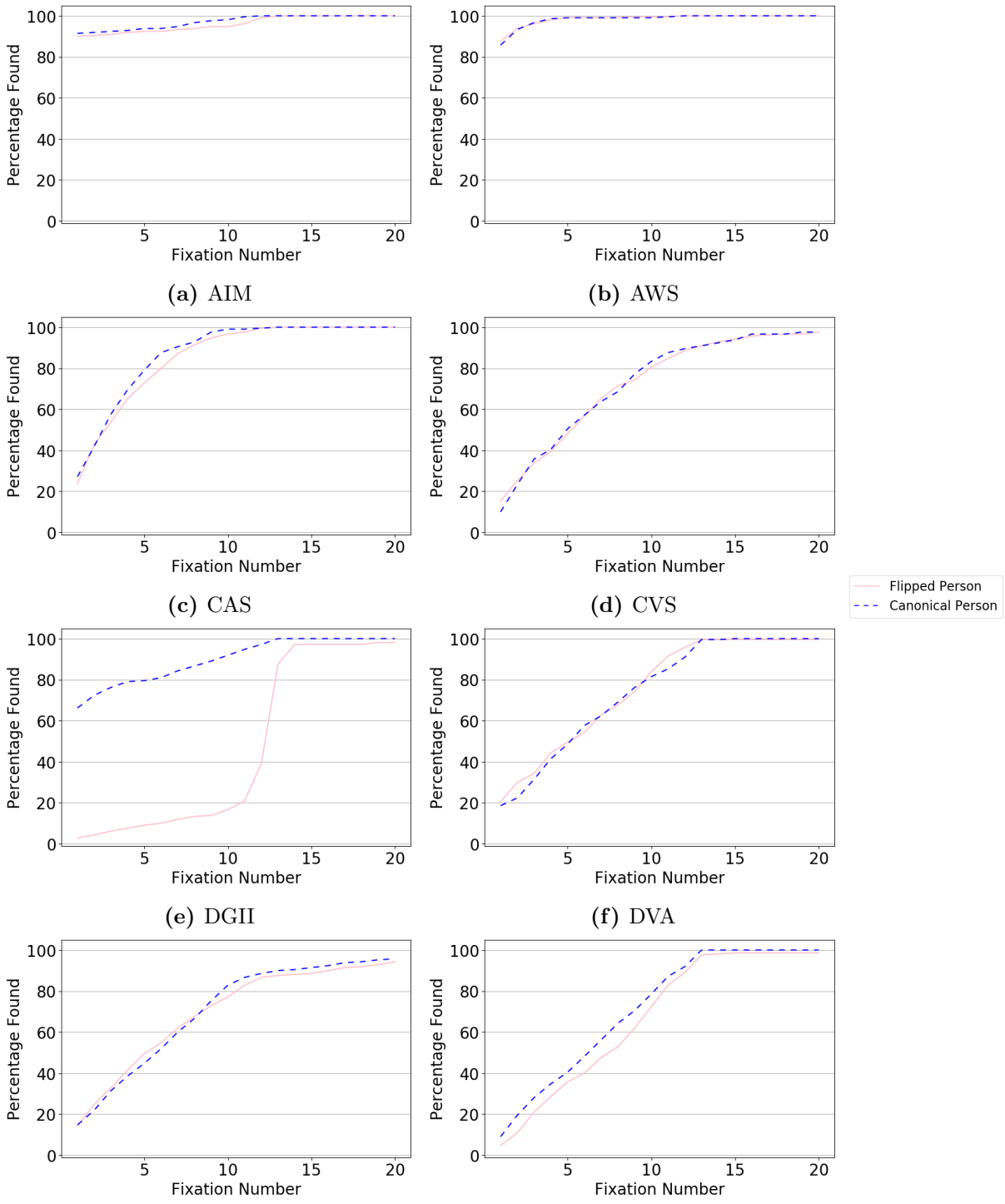
in Figures 4.41 and 4.42.

In this experimental condition, ICF is the only learning-based model that shows only a negligible degree of performance asymmetry that is much more on par with the performance of the classical models. As with several of the classic asymmetry conditions, ICF appears to assign each element in the search array a similar saliency value, and traversal through the array is therefore essentially element-wise random. SalGAN and SAM both show a moderate asymmetry with preference for canonically oriented targets, both finding the canonical targets faster and with fewer errors. DGII and oSALICON both display very strong preference for the canonically oriented targets, producing higher average and maximum salience ratios for canonically oriented targets as well as locating them more quickly on average than flipped targets.

Classical models, by contrast, show at most exceedingly small asymmetries in performance, with the majority of models showing no major difference in acquisition speeds for either target type. AIM, AWS, CAS, QSS, and SSR show relatively strong overall search performance, on average finding the target much faster than what would be expected by a random traversal. Most other classic models either exhibited a modest error rate (*e.g.* IKN) or failed to clearly distinguish targets from distractors (*e.g.* DVA). The only classical models to show statistically significant differences in saliency ratios are AIM (a very small preference for flipped mask targets in terms of both average and maximum ratios), IMSIG (a significant effect for canonical person targets and flipped mask targets only in the average ratio), and CVS (a significant preference for flipped person targets only

in the maximum ratio).

Overall, this experimental condition produces a relatively clear split in performance patterns between the classical and deep-learning based models. ICF, having a readout network trained in a fashion similar to the training method for DGII but explicitly not basing its predictions on deep features, in this case appears to be closer to the classical models in terms of its pattern of performance. For the methods that are based on deep features, the consistent asymmetry in performance is in the direction opposite that expected of human subjects.



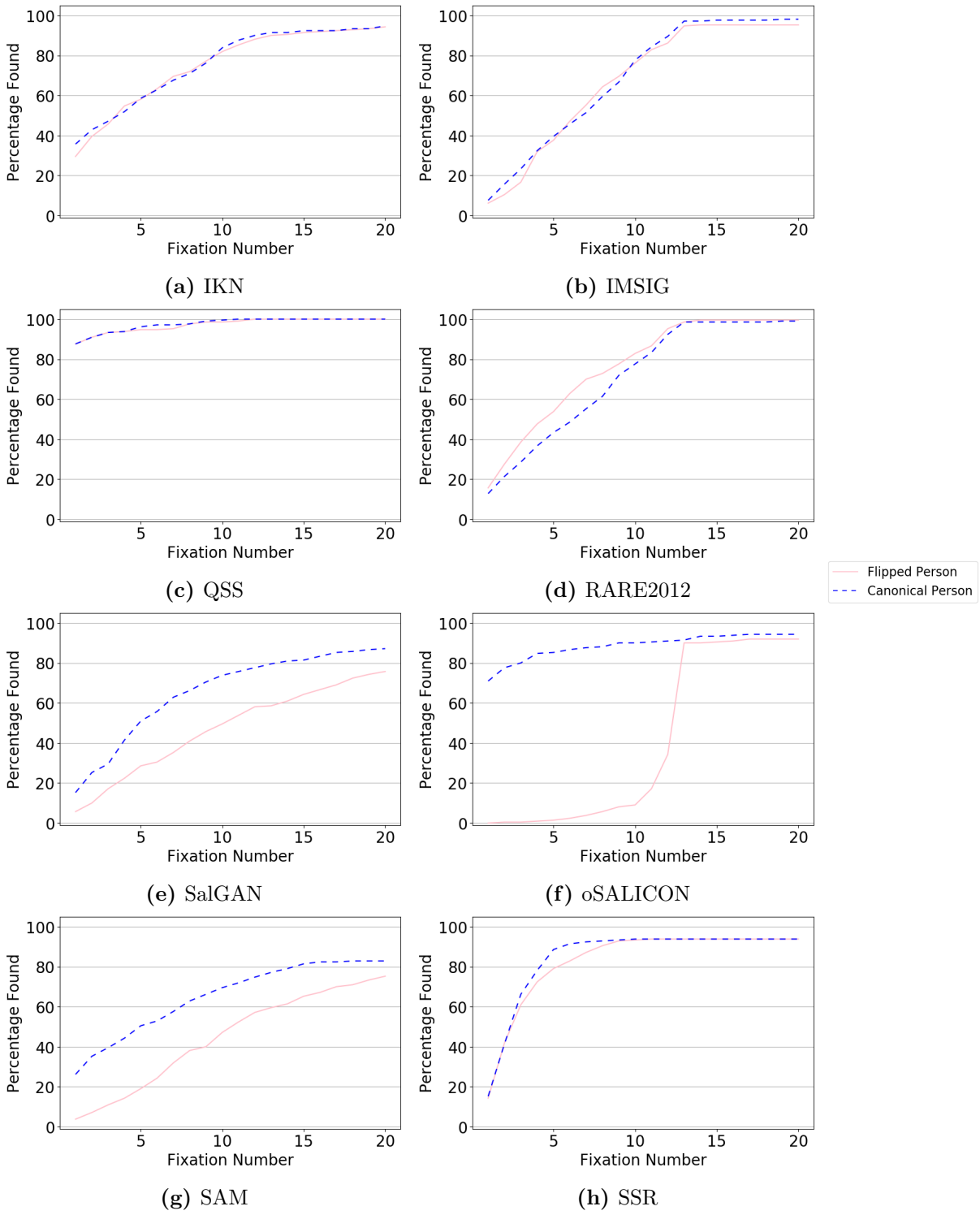
(a) AIM **(b)** AWS

(c) CAS **(d)** CVS

(e) DGII **(f)** DVA

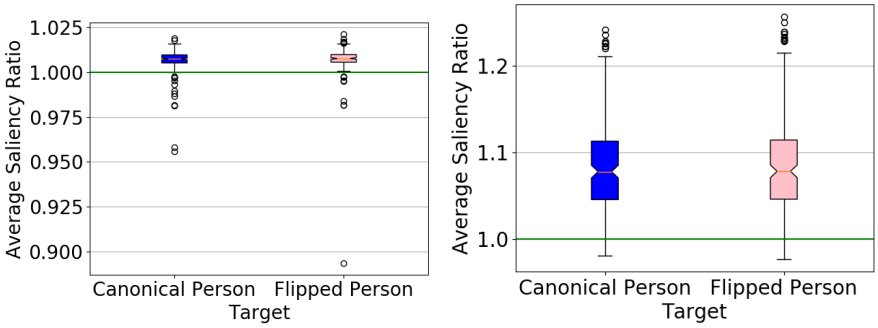
(g) GBVS **(h)** ICF

Figure 4.37: The cumulative probability of fixating the target for a given number of steps for models AIM-ICF when searching for a flipped person target amongst canonical person distractors (pink line) or when searching for a canonical person target amongst flipped person distractors (dashed blue line).



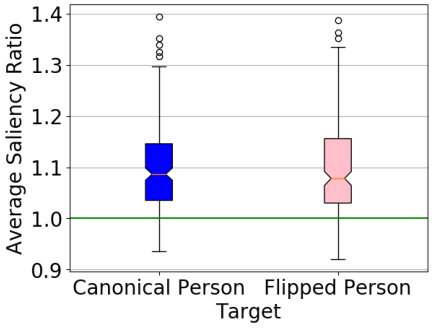
(a) IKN **(b)** IMSIG **(c)** QSS **(d)** RARE2012 **(e)** SalGAN **(f)** oSALICON **(g)** SAM **(h)** SSR

Figure 4.38: The cumulative probability of fixating the target for a given number of steps for models IKN-SSR when searching for a flipped person target amongst canonical person distractors (pink line) or when searching for a canonical person target amongst flipped person distractors (dashed blue line).

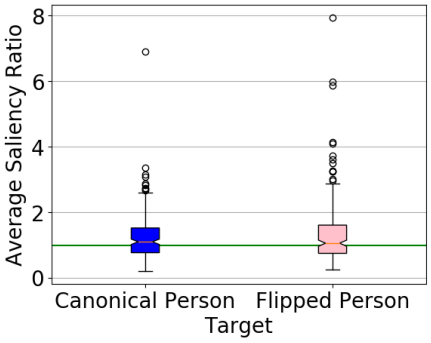


(a) AIM

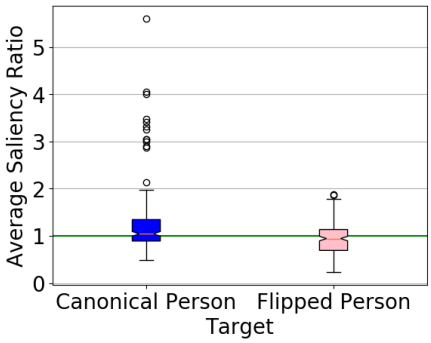
(b) AWS



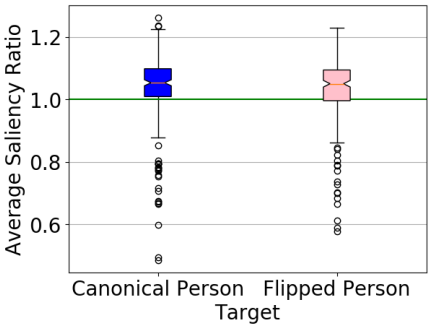
(c) CAS



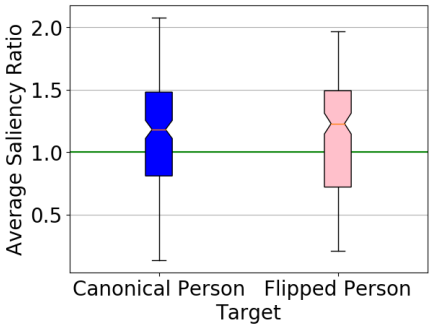
(d) CVS



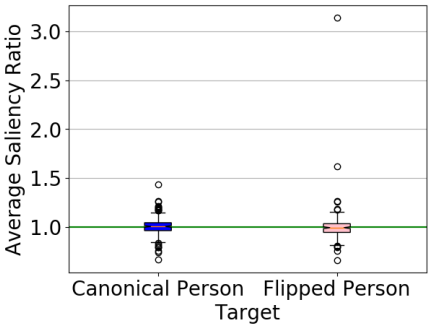
(e) DGII



(f) DVA

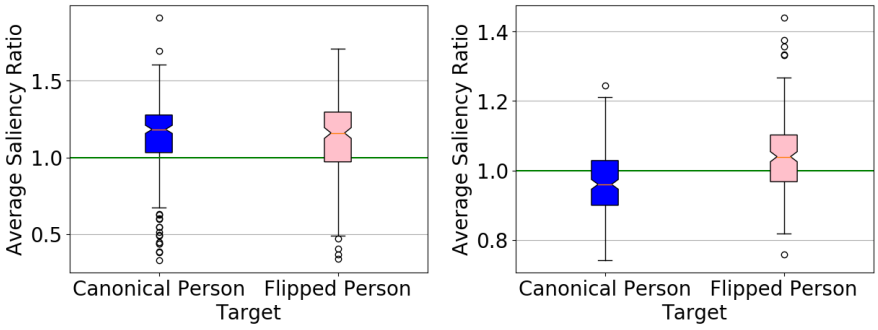


(g) GBVS



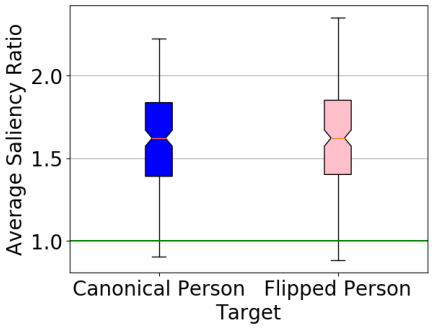
(h) ICF

Figure 4.39: The ratio of target to distractor average saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical person targets amongst flipped person distractors, whereas results on the right (pink) are for flipped person targets amongst canonical person distractors.

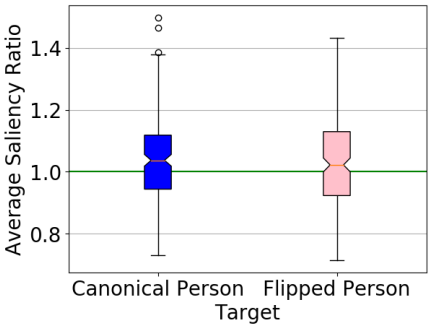


(a) IKN

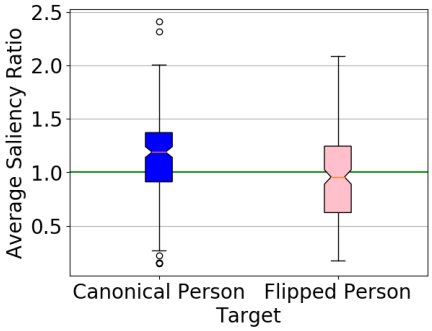
(b) IMSIG



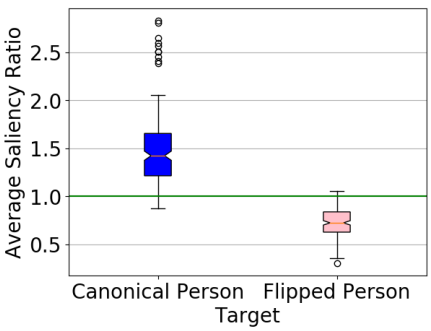
(c) QSS



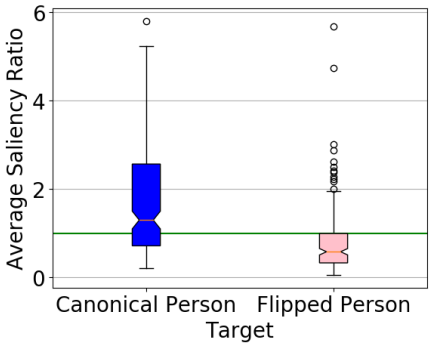
(d) RARE2012



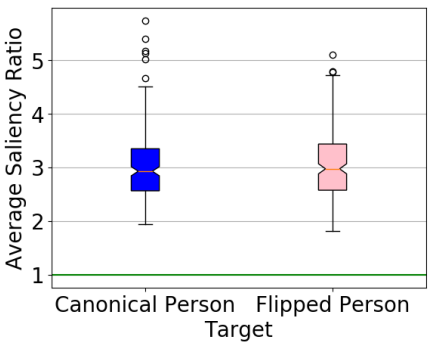
(e) SalGAN



(f) oSALICON

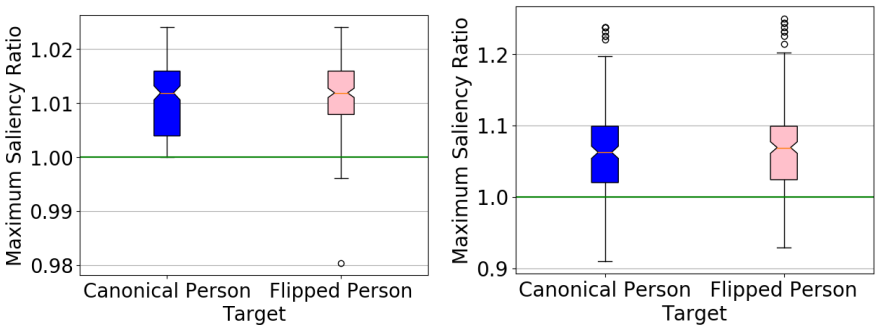


(g) SAM



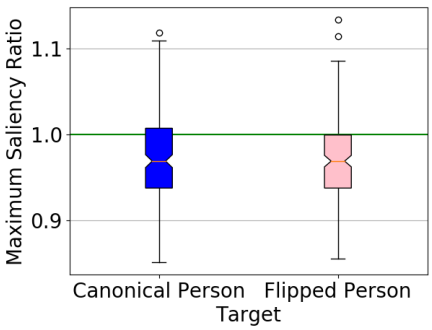
(h) SSR

Figure 4.40: The ratio of target to distractor average saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical person targets amongst flipped person distractors, whereas results on the right (pink) are for flipped person targets amongst canonical person distractors.

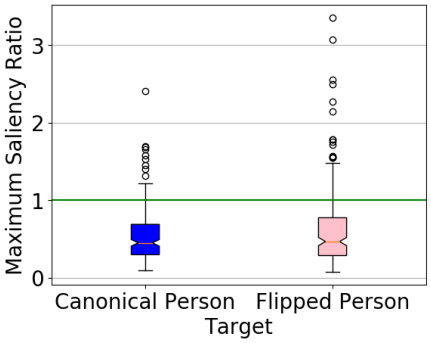


(a) AIM

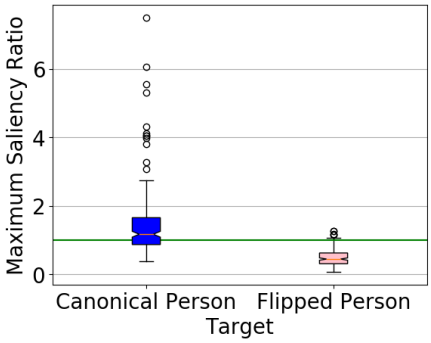
(b) AWS



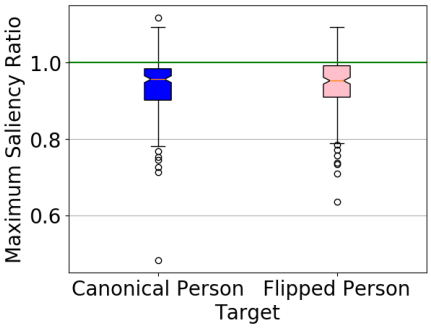
(c) CAS



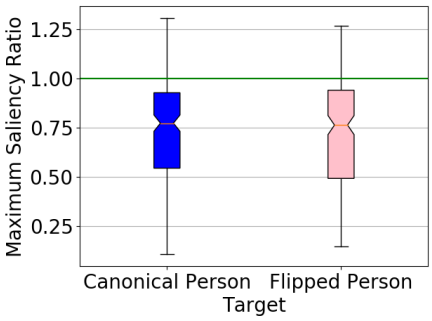
(d) CVS



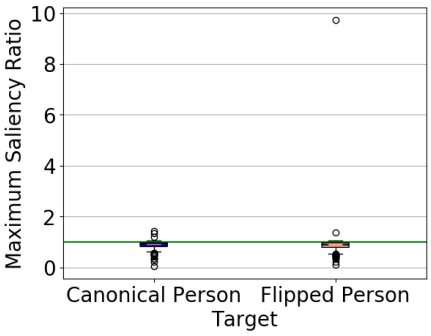
(e) DGII



(f) DVA

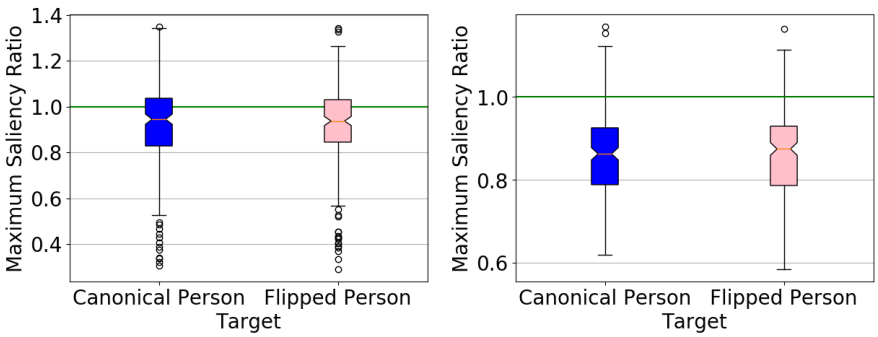


(g) GBVS



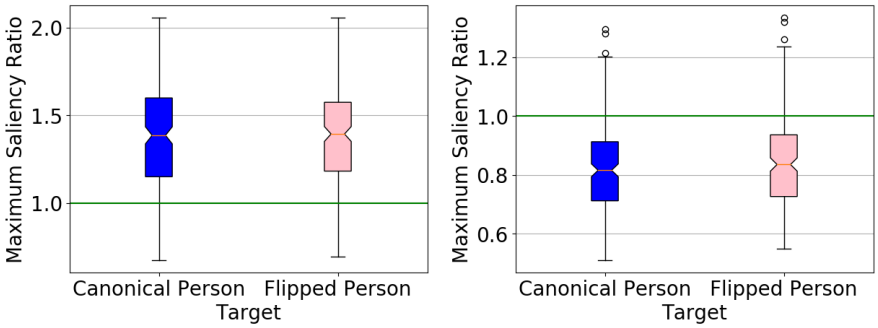
(h) ICF

Figure 4.41: The ratio of target to distractor maximum saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical person targets amongst flipped person distractors, whereas results on the right (pink) are for flipped person targets amongst canonical person distractors.



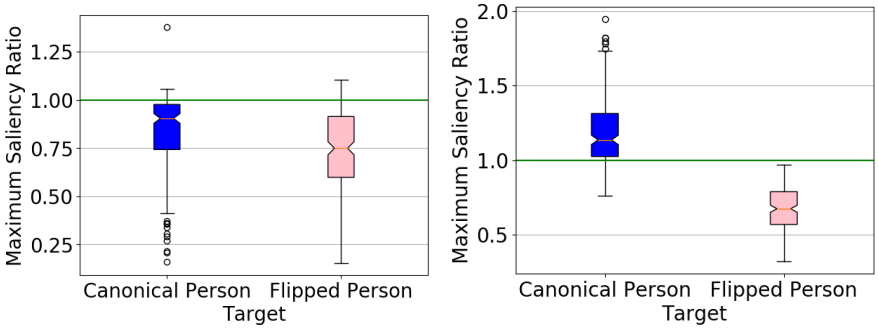
(a) IKN

(b) IMSIG



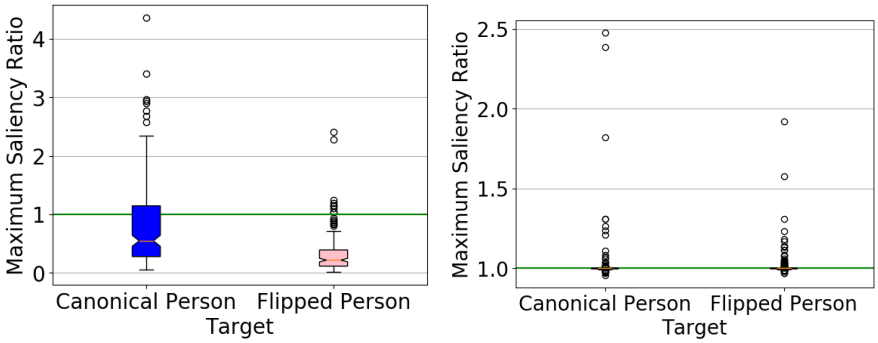
(c) QSS

(d) RARE2012



(e) SalGAN

(f) oSALICON



(g) SAM

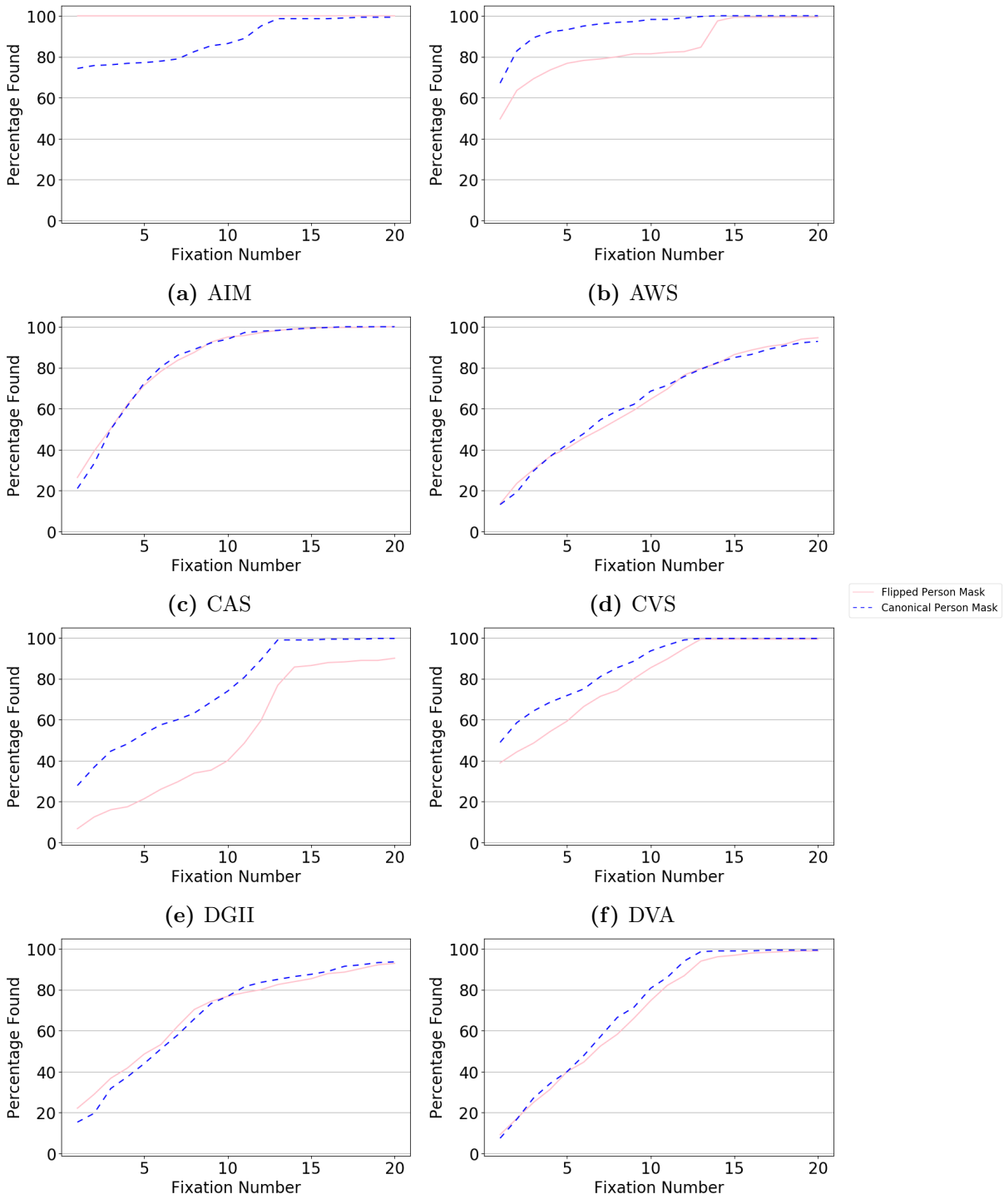
(h) SSR

Figure 4.42: The ratio of target to distractor maximum saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical person targets amongst flipped person distractors, whereas results on the right (pink) are for flipped person targets amongst canonical person distractors.

4.2.3.2.2 Flipped Mask vs. Canonical Mask: Similar to the fully detailed person arrays, when searching for a flipped or canonical mask the deep learning algorithms appear to uniformly favour the canonically oriented target mask (with the exception of ICF, which again shows negligable asymmetry). The effect size, however, is reduced, particularly for DGII and oSALICON that now show much more modest asymmetries (and, particularly for DGII, the saliency ratio measures show a much smaller asymmetry effect which is not significant in the average ratio measure). This could partly be explained, however, by the drop in performance for these models. Overall error rates on masks is distinctly higher than for fully detailed image elements. Since deep learning models are not as familiar with masked silhouettes, it is not necessarily surprising to see a smaller asymmetry effect.

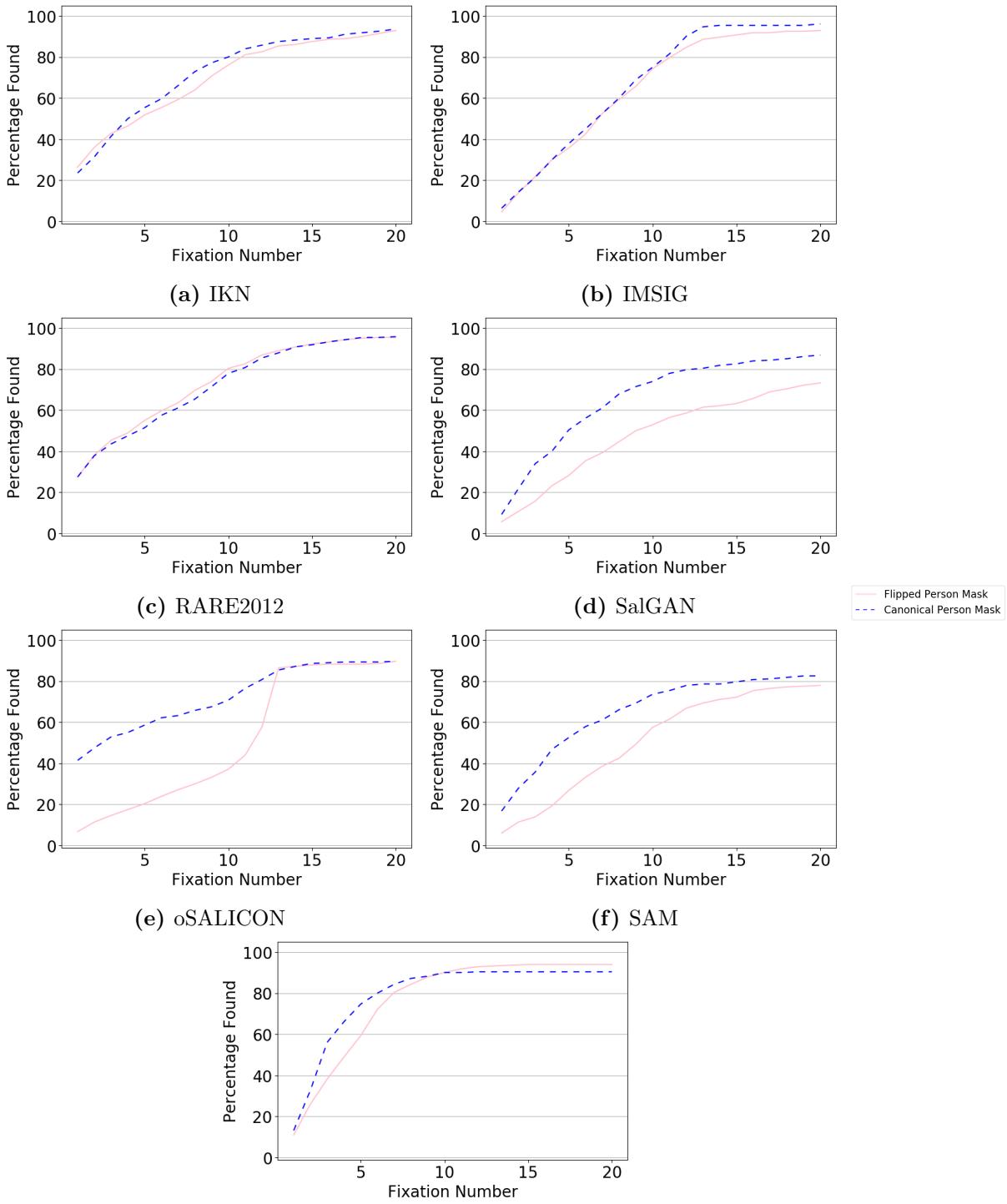
Somewhat surprisingly, there are several modest asymmetries found in the performance of some classical models for this condition. AIM, AWS, and DVA all show an asymmetry in performance, though it is not consistent in direction (*e.g.* AIM finds the flipped target first in every trial, whereas for canonical targets must search in a subset of trials, whereas AWS preferentially finds the canonical target). Furthermore, the saliency value ratios for these models suggest only minor numerical differences which are not significant leading to these performance patterns (particularly for AWS and DVA).

Overall this condition yielded some unexpected (though small) asymmetries in the behaviour of classical models, as well as more muted asymmetries in the deep models. Nevertheless, whereas the classical model asymmetries were inconsistent between models (with AIM showing asymmetric performance in line with that expected of humans, and AWS and DVA instead showing an inverse effect), the deep models again showed a consistent asymmetry in the opposite direction expected for humans.



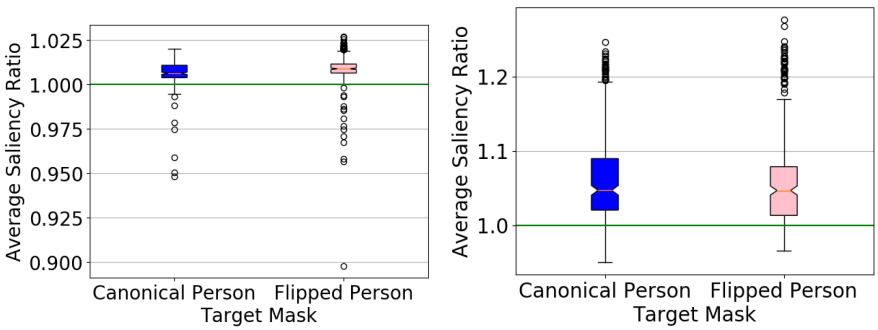
(a) AIM **(b)** AWS **(c)** CAS **(d)** CVS **(e)** DGII **(f)** DVA **(g)** GBVS **(h)** ICF

Figure 4.43: The cumulative probability of fixating the target for a given number of steps for models AIM-ICF when searching for a flipped person mask target amongst canonical person mask distractors (pink line) or when searching for a canonical person mask target amongst flipped person mask distractors (dashed blue line).



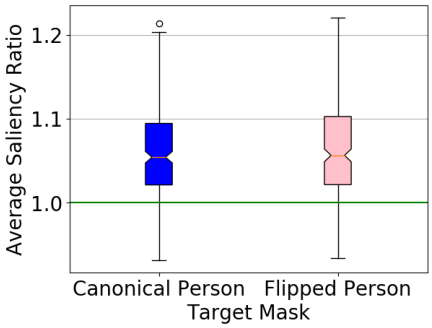
(a) IKN **(b)** IMSIG **(c)** RARE2012 **(d)** SalGAN **(e)** oSALICON **(f)** SAM **(g)** SSR

Figure 4.44: The cumulative probability of fixating the target for a given number of steps for models IKN-SSR when searching for a flipped person mask target amongst canonical person mask distractors (pink line) or when searching for a canonical person mask target amongst flipped person mask distractors (dashed blue line).

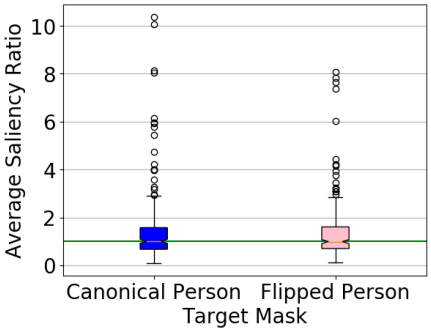


(a) AIM

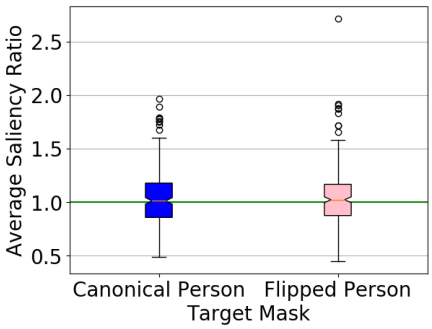
(b) AWS



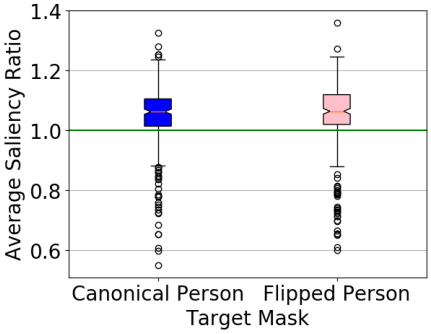
(c) CAS



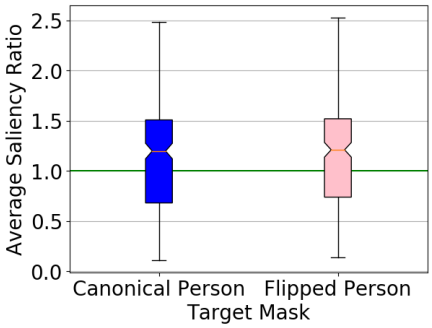
(d) CVS



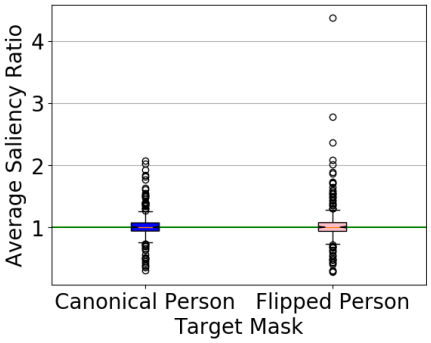
(e) DGII



(f) DVA

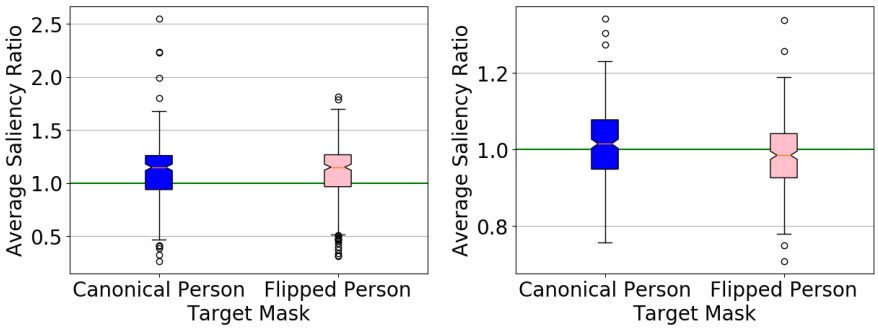


(g) GBVS



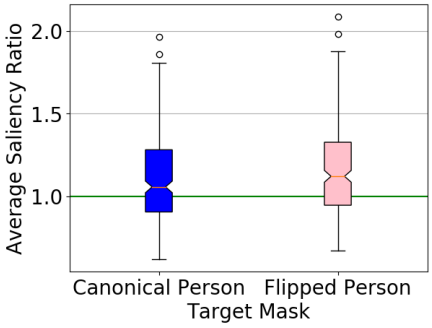
(h) ICF

Figure 4.45: The ratio of target to distractor average saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical person mask targets amongst flipped person mask distractors, whereas results on the right (pink) are for flipped person mask targets amongst canonical person mask distractors.

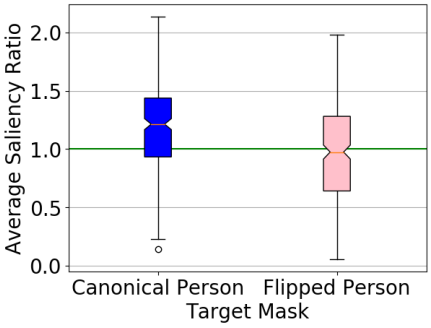


(a) IKN

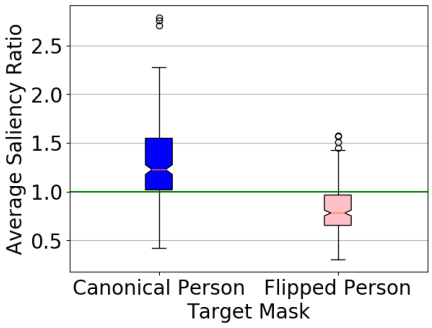
(b) IMSIG



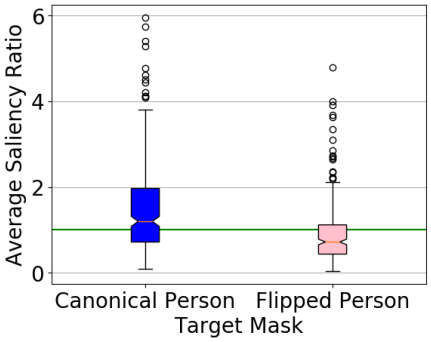
(c) RARE2012



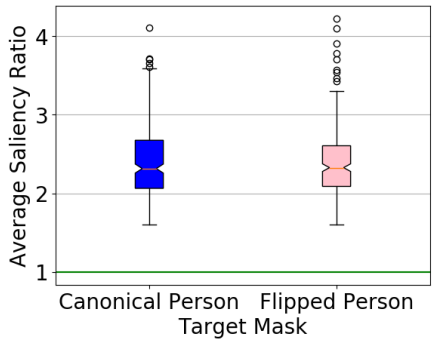
(d) SalGAN



(e) oSALICON

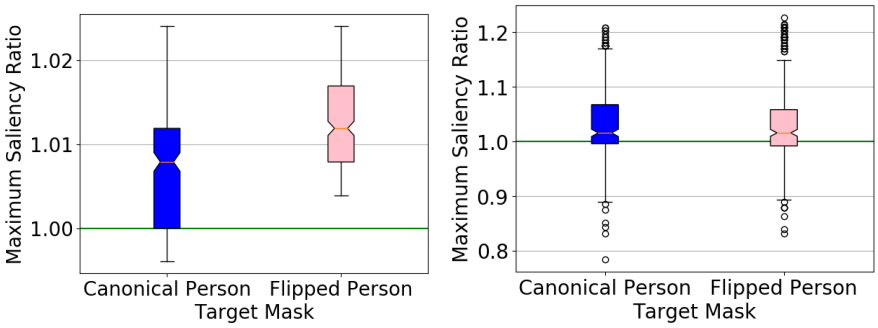


(f) SAM



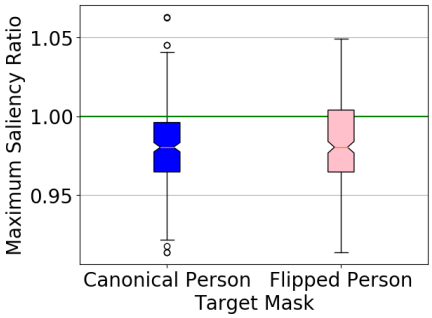
(g) SSR

Figure 4.46: The ratio of target to distractor average saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical person mask targets amongst flipped person mask distractors, whereas results on the right (pink) are for flipped person mask targets amongst canonical person mask distractors.

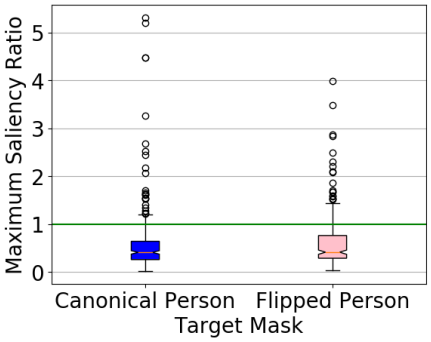


(a) AIM

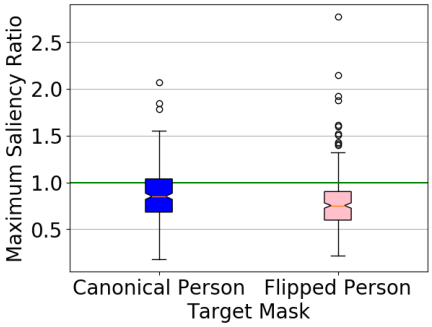
(b) AWS



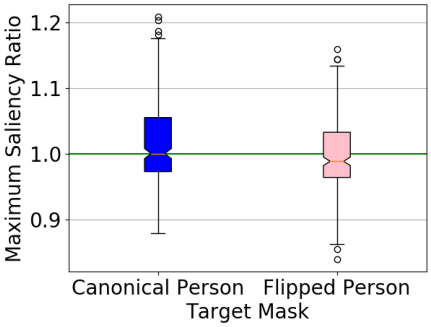
(c) CAS



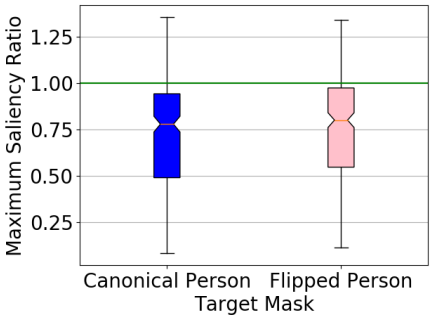
(d) CVS



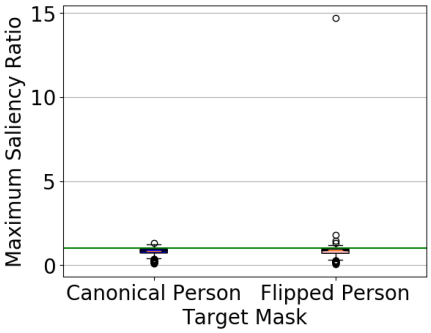
(e) DGII



(f) DVA

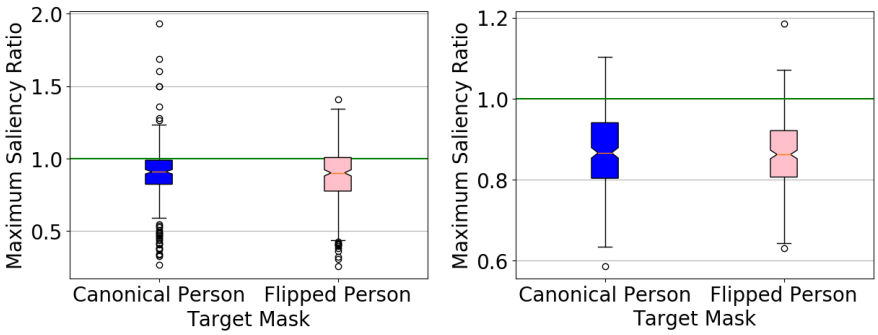


(g) GBVS



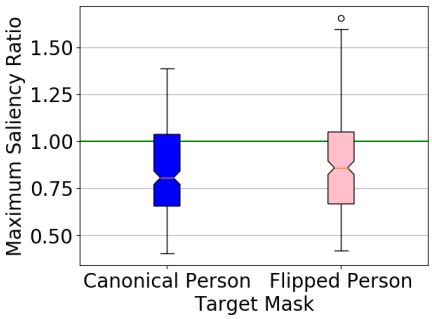
(h) ICF

Figure 4.47: The ratio of target to distractor maximum saliency values for AIM through ICF. Results on the left in each plot (blue) are for canonical person mask targets amongst flipped person mask distractors, whereas results on the right (pink) are for flipped person mask targets amongst canonical person mask distractors.

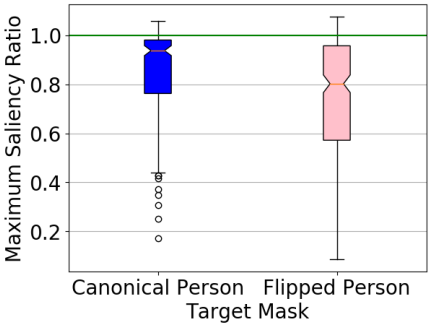


(a) IKN

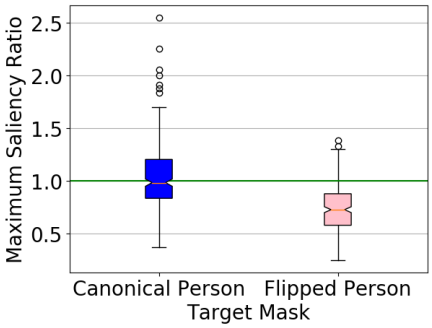
(b) IMSIG



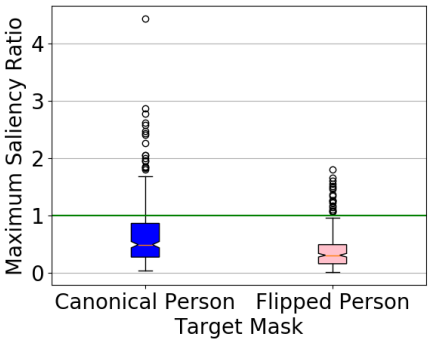
(c) RARE2012



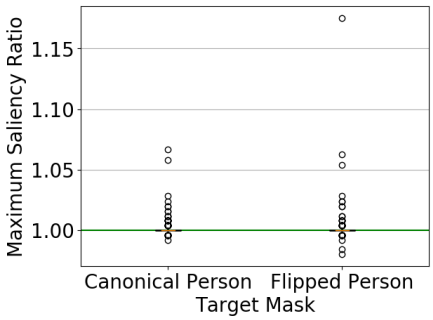
(d) SalGAN



(e) oSALICON



(f) SAM



(g) SSR

Figure 4.48: The ratio of target to distractor maximum saliency values for IKN through SSR. Results on the left in each plot (blue) are for canonical person mask targets amongst flipped person mask distractors, whereas results on the right (pink) are for flipped person mask targets amongst canonical person mask distractors.

4.2.4 Discussion

As with the orientation experiment presented in Section 4.1, performance in the classic asymmetries in Section 4.2.3.1 was highly variable between models, not just in reproducing the predicted search asymmetries found in human subjects, but in even being able to accomplish the visual search task and effectively mark the target as sufficiently salient to locate it in a rank order traversal of the map. Although many classic models struggled with the different experimental conditions, there were several that showed overall quite strong performance such as AIM, AWS, CAS, DVA, and IMSIG. Although not always able to accurately predict the expected asymmetry or display a target acquisition rate which could be expected from human observers (for example, AIM showed a clear preference for blue dots over magenta dots, even in a magenta singleton condition), these four algorithms nevertheless were able to consistently locate the target and accurately predicted some of the expected asymmetries. RARE2012 had trouble with the colour singleton test, but otherwise presented strong performance. QSS, likewise, struggled to process certain stimuli classes, but otherwise showed a comparatively solid performance for those conditions in which it could operate.

The deep learning models, despite their overwhelming performance in standard fixation prediction benchmarks, largely struggled with this task. oSALICON showed the strongest performance, but still presented a small error rate and for some of the conditions (such as the flipped A and canonically oriented A) failed to distinguish the target from the distractors in any meaningful way (successfully locating the target by a linear search of scene elements). This overall poor performance on the part of deep learning models suggests that there is a mismatch between the focus on fixation prediction benchmarks and a more general reflection of human representations of salience. In particular, the models with more complex deep learning techniques such as the use of Generative Adversarial Networks (GANs) or Long Short-Term Memory (LSTM) units (SalGAN and SAM, respectively) appear to suffer more greatly in the domain shift, possibly suggesting that these techniques are poorly suited to the problem of more general saliency modelling. It is also possible, however, that the design of these models enforces a centre bias component throughout the saliency calculation (as opposed to a spatial prior applied in post-processing, which can be easily excised), and this alone is responsible for their poor performance on search arrays that lack a biased target

distribution.

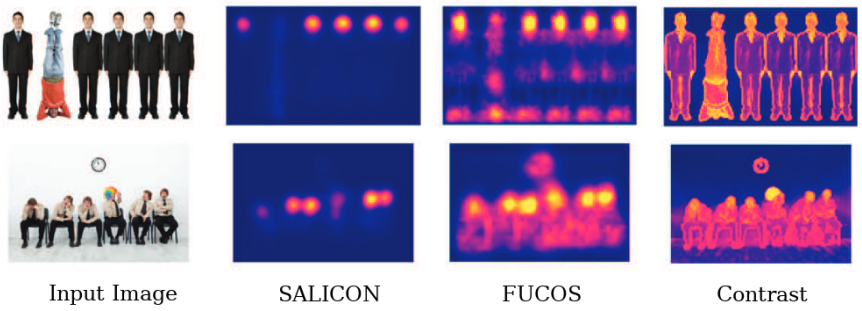


Figure 4.49: Example natural images which challenge deep learning models of saliency, showing resultant saliency maps produced by two deep learning based models (oSALICON [7] and FUCOS [8]), and a feature contrast-based model [9]. Image reproduced from [8]

In the complex object case, both classical and deep models show a reduction in miss rate. Examining the rate of target acquisition, however, reveals that all models based on deep features predict a bias in the opposite direction than expected for human observers. Previously, Bruce *et al.* [8] have noted images for which deep learning models are outperformed by more classical models, including the example images shown in Figure 4.49. However, while Bruce *et al.* focused their explanation on feature contrast as a reason for the failure of the deep models to explain the failure of the deep models to adequately highlight both the upside down man in the bright clothing and the fellow in clown hair, it is possible that the issues go deeper than this.

There have been a number of prior framings of saliency that have acknowledged the role of surprise or predictability in attentional capture [211, 2, 159, 155, 267], pointing out that scene elements which are unexpected or unusual stand out as salient and attract an observer's attention. The experiential or knowledge-based aspect of this formulation for salience is also consistent with the explanation Bruce and Tsotsos [156] put forward for "novelty" based asymmetric search results, as novel stimulus elements are surprising and consequently lack a compact neuronal representation, leading to them asymmetrically standing out in comparison to the more familiar scene elements.

Deep learning is essentially a very powerful system for function fitting, using a complex hierarchy of convolutions and non-linearities to map an input X onto some output Y based on a large corpus of training data. This is a powerful tool for many vision problems, and allows a model to, given sufficient high quality data, learn the output necessary to provide a strong prediction for a specific

problem domain. In the case of saliency models, these techniques have focused on learning what scene elements attract human fixations in natural images, and do so remarkably well. However, the output that is learned best is the output that is best represented by the training data. Humans in a scene will tend to attract observer fixations, and humans usually appear in their canonical orientation. A deep learning saliency model for fixation prediction, therefore, will likely place the most emphasis on human targets *as they most frequently appear*, even though a human in an unusual state (such as upside down) should actually be more salient. Unusual or surprising scene elements are, by their very nature, rare, and are thus very difficult to learn in a data-oriented deep learning approach.

A second note regarding deep models of saliency is how they deal with within-scene surprise. Contextual and element comparison is a large part of identifying salient targets, whether in terms of oriented bars, colours, or other singletons. However, Ricci *et al.* [268] recently showed that feedforward networks struggle with the ability to learn visual relationships (specifically in their case a same-different judgement), and that once intra-class variability exceeded network capacity the networks failed completely. More research would likely need to be done to see if singleton detection or other psychophysical judgements of saliency can be effectively learned (as the networks tested here were admittedly designed and tested for natural images), but this preliminary research into visual relationships suggests that feedforward deep networks may not be suited to these types of contextual problems.

4.3 Temporal Order and Saliency

A third case study shifts away from psychophysical tests of saliency models and returns focus to fixation prediction. This study showcases SMILER's ability to facilitate the study of general characteristics of saliency model performance across multiple models, helping to increase the robustness and generality of conclusions. The particular topic of this study is a return to the question of how strongly saliency correlates with human fixations through time.

4.3.1 Motivation

A major question and source of enduring debate within saliency research is on how great a role target salience makes in the decision to fixate said target. Given the prominent role that fixation data serves in the evaluation of saliency algorithms, this is an important question to answer. Nevertheless, outlooks on the topic run the gamut from the explicit argument that saliency is the primary guide for fixation selection [64] and the implicit perspective that saliency algorithms can (and should, if sufficiently well developed) account for human fixation locations at least as well as one human's fixations predict another's [260], through to the suggestion that saliency plays a minimal role that is largely overridden by task guidance and scene understanding [112, 113]. Much of this tension is driven by different implicit assumptions about the generality of claims and experimental settings (see Section 1.2.1), but even within the condition for which saliency maps are most extensively designed for and show the strongest performance (free-viewing over natural scenes), the question arises as to whether there may be a temporal aspect to the role of saliency.

It is clear for dynamic stimuli that human observers, even without explicit task instructions, will naturally build a narrative understanding of the visual sequence that guides fixation targeting [127], so it is possible that a similar albeit likely weaker process occurs when viewing a static scene. With each fixation observers place over a scene they are increasing their understanding of the scene contents, and it thus is possible that as this higher level understanding develops it begins to take greater precedence over saliency as a guide for fixation selection. Parkhurst *et al.* [96] found such a correlation, with earlier saccades having higher saliency values than subsequent saccades until average saliency values leveled off after the fifth saccade. Underwoood *et al.* [269] found a similar pattern when comparing the cumulative probability of fixation for high and low salient objects in a natural scene, though this effect disappeared when observers were given explicit target instructions.

There are, however, a number of potential issues with this claim. Tatler *et al.* [95] performed a similar analysis to Parkurst *et al.* [96] and found that while the contribution of saliency does indeed reduce through time, this appears to be a product of the compositional bias of their image set toward central regions having higher average salience. Since earlier fixations by human observers are allocated toward the centre of an image (irrespective of starting location [193]), this would end up conflating the measure of saliency through time. Additionally, in dynamic stimuli it is hypothesized

that it is higher-level considerations that are predominantly guiding fixations, but this tends to make inter-subject fixation targeting *more* consistent [124], whereas for human observers over static stimuli inter-subject consistency decreases through time [95]. It is possible that dynamic stimuli provide an implicit task structure through the desire to follow the sequence of events unfolding, whereas this is either lacking or rapidly understood for static scenes, leaving subjects free to base subsequent decisions on individual considerations. Nevertheless, if higher level considerations do become more prominent through time and decrease the impact of saliency on fixation selection when viewing static stimuli, it would at least appear to be doing so in a different manner than is seen in dynamic stimuli.

Some more recent studies have approached the topic from a different direction, electing to manipulate the stimuli rather than rely on passive correlation of fixation locations and saliency values. This methodology helps to avoid confounds based on the image composition, as the same spatial layout can be used with different distributions of stimulus conspicuity, thereby ensuring that any shifts in fixation of the manipulated region were driven by the stimulus properties rather than the global composition of the image. Anderson *et al.* [270] used a set of natural images in which a part of the visual field had the contrast manipulated to be either increased or decreased. In both a free exploration condition and a target search condition, it was found that there was a marked increase in first saccades landing on a region of high salience. De Vries *et al.* [271] focused on the question of whether the perceptual draw of high salience existed only for the first saccade or if it could persist for subsequent saccades. They used a grid of oriented bars with annuli of varying luminance placed around three targets in order to manipulate the apparent saliency of the target elements. While the annuli denoted potential targets with high conspicuity, the status of the elements (target or distractor) was not discernable peripherally, saccades were predominantly driven by the salience of the annuli. However, when the grid object sizes were increased to allow for peripheral discernability, the saliency contribution of the annuli no longer seemed to influence saccades, consistent with the results of Underwood *et al.* [269] in which target instructions appear to override considerations based purely on stimulus conspicuity.

The question of whether considerations of saliency for fixation targeting are greater during initial scene viewing or consistent through time is therefore unclear. All evidence points toward top-down task guidance (such as being given a search target) as having a powerful confounding

effect on an observers's reliance on saliency, and it is at least reasonable to expect that the chance of an observer selecting an intrinsic task driver will increase through time. Additionally, the results of De Vries *et al*. [271] suggest that an additional confounding factor is the degree of uncertainty in a location's information content (as predicted by a visibility model [40], discussed in more detail in Section 1.2.2.6), which could also help explain why the effect appears to be obscured by a centre bias as early central fixations are an effective strategy for maximally reducing overall uncertainty in scene content [193]. Nevertheless, in all of the studies that have attempted to elucidate the role of saliency through time, the saliency representation is typically represented by only one model (in [96, 95, 269]) or stimulus attribute (such as contrast [270] or luminance [271]). It is possible that perhaps different models, each of which represents saliency in a different manner, will correlate differently with fixation selection through time. Seeing the temporal pattern of fixation saliency for a variety of models may allow us to better understand how robust a time-dependent model of saliency relevance is, and possibly even determine whether there are different time courses for different features in the representation of salience.

4.3.2 Method

In order to study the relationship of fixation temporal order to saliency prediction, the CAT2000 dataset[12] provides a useful corpus of fixation data. Containing two thousand images organized into twenty different categories, this dataset provides a larger quantity of fixation data than previous studies on this topic while also allowing the investigation of whether the results are impacted by the image category (such as social settings, outdoor settings (both natural and man-made), or noise-degraded images). One of the other reasons why CAT2000 is an appropriate choice for this study is that it is released with individual subjects' temporally ordered fixations available. However, a number of fixations included in the individual sequences of observers were outside the bounds of the image. In order to prevent spurious comparisons with out of bound fixations while still ensuring cohesive sequences, the CAT2000 data was groomed by truncating any sequence that went out of bounds to the final in-bounds fixation location. If this truncation left the sequence with fewer than ten total fixations, it was discarded completely. Of 36000 total recorded fixation sequences, this criterion led to the elimination of 6257 sequences.

In Section 4.3.3, CAT2000 is first analyzed for potential systemic confounds, replicating Tatler

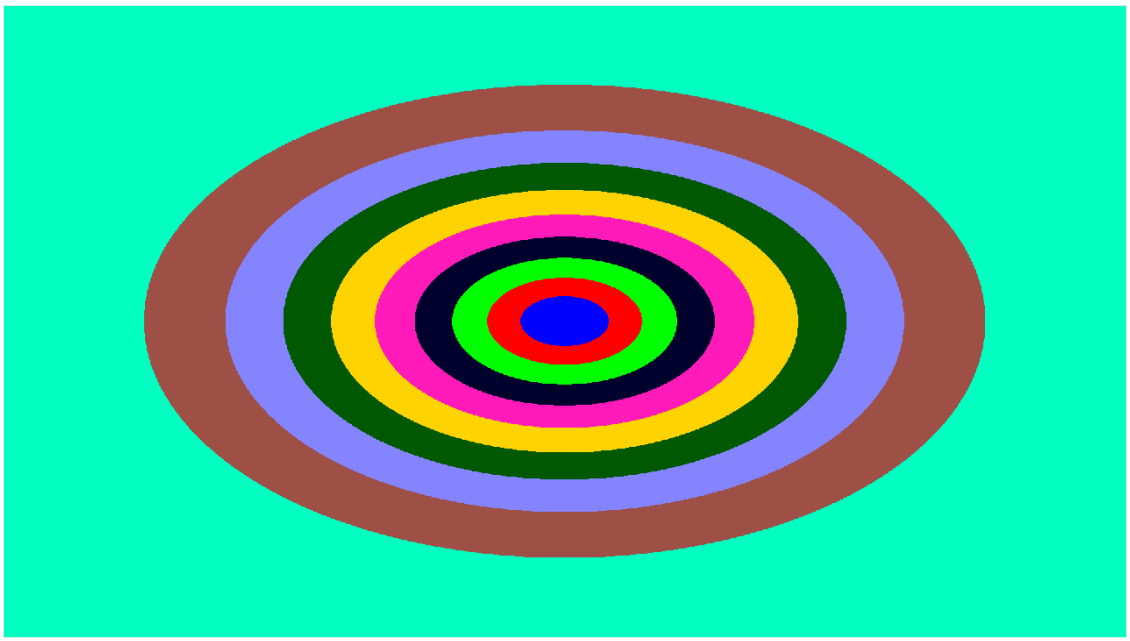


Figure 4.50: Spatial regions in the CAT2000 dataset with an equal number of fixations falling into each bin

et al.'s [95] findings that earlier fixations tend to be more centralized and more consistent among subjects. In order to deal with the potential confound of a centre bias in correlating fixation temporal order with salience value, the methodology of Chapter 2 is adopted to create a set of spatiotemporal bins B_st, where subscript s denotes the spatial bin (starting with bin 1 at the centre of the image) and t denotes the temporal place in a fixation sequence. As with Chapter 2, images are split into ten spatial regions, as shown in Figure 4.50.

Saliency maps were generated using SMILER for the following algorithms: AIM [2], AWS [162], CAS [143], CVS [145], DGII [182], DVA [160], GBVS [78], ICF [184], IKN [25], IMSIG [168], QSS [167], RARE2012 [148], SalGAN [189], oSALICON [7], SAM-VGG [186], and SSR [161]. All algorithms were set to use default parameters and smoothing values (including centre biasing), and YAML files which can be used to generate the saliency maps analyzed in this experiment can be found in Appendix A.3.

Scores were assigned for each spatiotemporal bin according to two metrics: NSS and AUC. The inclusion of NSS is intended to allow for more direct comparison to the work of Parkhurst *et al.* [96], who compared raw saliency scores for fixated pixels across time. Although NSS includes a normalization step to better compare values between algorithms, it should be noted that it is

still difficult to directly compare scores across algorithms as the normalized map scores will be influenced by the sparsity of the original maps, a characteristic which can vary widely between algorithms. Therefore, AUC is also included, as it provides an evaluative score based on pixel rank-order independent of the specific numerical spread of the algorithms. The results of this analysis are presented in Section 4.3.4.

4.3.3 CAT2000 Statistics

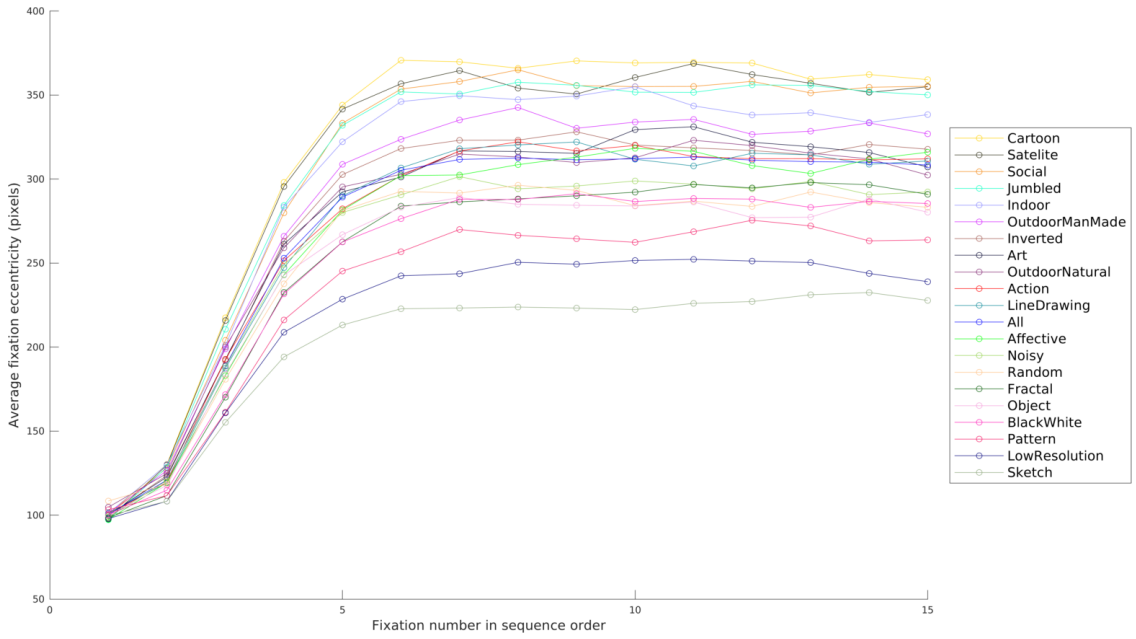
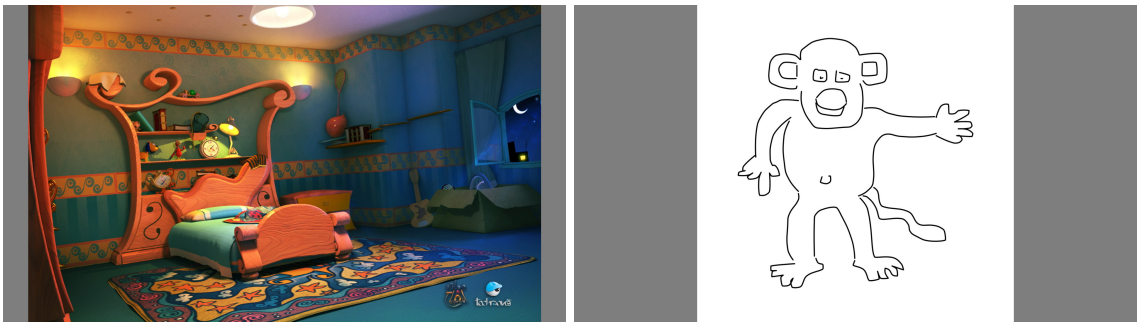


Figure 4.51: Average fixation eccentricity (in pixels from the image centre) by fixation temporal order for each category in CAT2000. As can be seen, fixations tend to move further from the centre during the first several fixations before appearing to saturate after the fifth fixation. Category labels are given in descending order of area under the curve.

Figure 4.51 shows the average fixation eccentricity (measured in pixels from the image centre) of human fixations as a function of the fixation sequence order. Results are shown separately for each category, with the average over all categories shown as the "All" line in blue. There is clearly a strong temporal effect across all categories that follows a very similar pattern of saturation around the fifth fixation, matching the results of Tatler *et al.* [95]. While the overall pattern of fixation eccentricity is the same across categories, the magnitude of the spread appears to be strongly affected by the category composition. Interestingly, the standout categories in both extremes are

formed by artificial stimuli, with the "Cartoon" category displaying the greatest average eccentricity of fixation and the "Sketch" category showing the most centrally clustered fixations. Example images from these two categories are displayed in Figure 4.52; as can be seen, the Cartoon images are largely composed of detailed and busy scenes with many elements to explore, whereas the Sketch images are localized to a smaller, centralized canvas with only sparse detailing.



(a) Cartoon **(b)** Sketch

Figure 4.52: Example images from the Cartoon (left) and Sketch (right) categories of the CAT2000 dataset. As can be seen, although both categories contain artificial stimuli, the nature of the stimuli in each is quite different. The Cartoon images frequently contain busy scenes with high levels of detail and bright colours, whereas the Sketch category contains a centered and rather sparse abstract representation of an object.

This compositional pattern is similar for the other categories on each end of the spectrum. "Satelite"[1] images contain large-scale aerial views with scene details well distributed throughout the image, whereas "Pattern" images are frequently very feature sparse. "LowResolution" is an interesting case, in that it is not composed of sparsely detailed artificial stimuli like the "Pattern" and "Sketch" categories, but is rather distinguished as a collection of heavily blurred images, whether natural or artificial. It would appear that the blurring of the images and reduction of the high frequency feature space ends up shifting the average saccade eccentricity toward the image centre, perhaps because the already reduced visual acuity of peripheral vision is more drastically affected by the loss in visual clarity due to blurring.

Figure 4.53 shows the average inter-subject agreement between saccade targets as a function of saccade temporal order. It is worth briefly discussing the metric used for inter-subject agreement and how it relates to previous analyses. In this study inter-subject agreement was calculated as

[1]Note that this is the spelling used within the dataset itself, rather than the correctly spelled "Satellite".

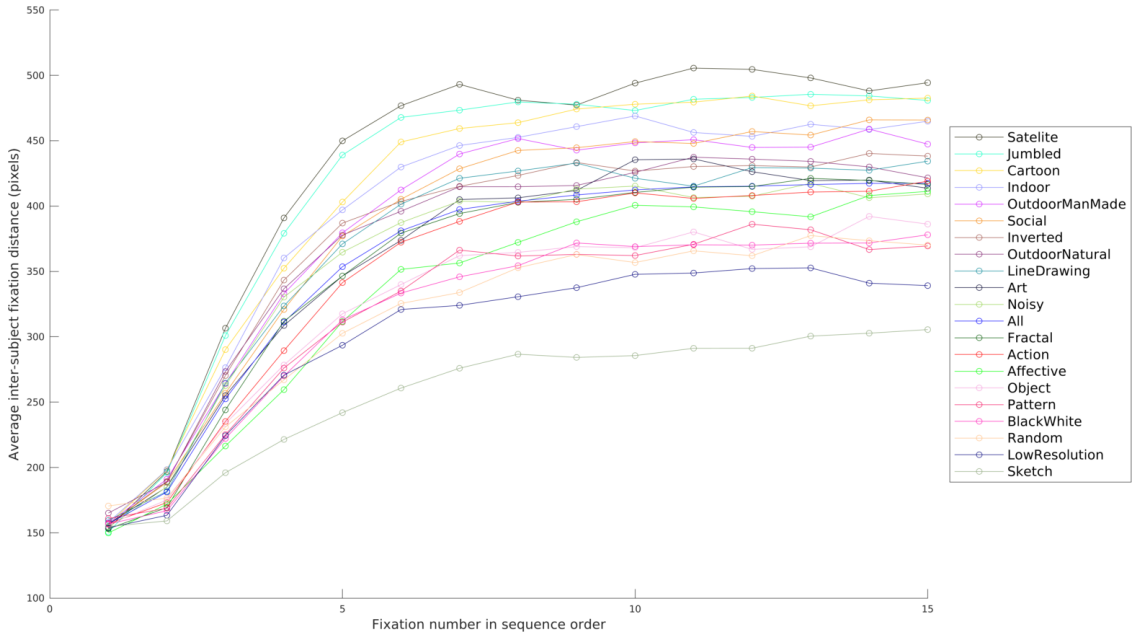


Figure 4.53: Average inter-fixation distance (in pixels) between observers for each fixation in temporal order for each category in CAT2000. As can be seen, inter-observer consistency tends to decline over time, saturating for most categories somewhere between the fifth and eighth fixation. Category labels are given in descending order of area under the curve.

the Euclidean distance between ordered pairs of fixations points, a common measure of trajectory similarity [272]. Mannan *et al.* [273] use a similar measure, although they do not preserve the temporal ordering of the fixation sequence and instead assign a distance value for each fixation based on the closest fixation in the sequence being used for comparison. Tatler *et al.* [95] object to this metric based on the argument that fixations should either be classified as the same or different, and that there really is no reason to classify a fixation that is 20° away twice as different as one that is 10° away. While this is a valid point when considering only the foveation of targets, it introduces the need to parametrically assign a function that classifies fixations as either matching or not. Tatler *et al.* do so by assigning fixational probabilities to 2° bins over the image and calculating the KL divergence between the probability maps created for human observers.

However, while certainly the location of greatest visual acuity, the fovea is not the only location from which visual information is sampled, and the informativeness of that visual sampling appears to vary continuously across the visual field [274]. This is the basis for the formulation of visibility models, and from this perspective there *is* a difference between a fixation that is 20° away

versus one that is 10° away. Although this relationship is unlikely to be a linear one, Euclidean Distance does provide a first-order approximation that is intuitively clear and obviates the need for a parameterization of the metric function.

Interestingly, although the inter-subject difference is measured with rather different methods, the results presented here find a remarkably similar pattern as those of Tatler *et al*. [95]. For all categories inter-subject agreement drops with fixation number before appearing to saturate somewhere between the fifth and eighth fixation. It is perhaps not particularly surprising that the magnitude of the effect across categories is quite similar to that found in Figure 4.51, as one would expect that a category for which fixations are more far flung will have a greater inter-subject distance between fixated locations should the subjects focus on different image regions. Nevertheless, it is interesting to note some of the changes in the order that is seen, such as the "Cartoon" category being pushed out of the top spot by both "Satelite" and "Jumbled". This is possibly due to the fact that the "Cartoon" images contain on average more semantic content that could drive fixations; the "Satelite" images are by their nature more homogenous in appearance, whereas the semantic content of the "Jumbled" images has been disrupted by the shuffling of image regions.

As saliency models for predicting fixation locations grow increasingly reliant on machine learning based on human data, it is important to keep in mind this shift in inter-observer consistency through time. Both the number of observers for whom data is gathered as well as the duration for which fixations are tracked for each image will end up having a large effect on the type of data being utilized for training. Much as a number of seemingly innocuous choices in the numerical and parametric choices in saliency map values and metric formulation can have profound impacts on the ranking of model performances [62], it is likely that the similarity in training and test data in terms of acquisition techniques (particularly in terms of number of subjects and sequence lengths) will have a large yet hidden role in model performance.

4.3.4 Results

4.3.4.1 Average NSS Scores Over All Categories

Temporally ordered NSS scores calculated by spatial bin over all categories are plotted for each algorithm, split into two groups based on alphabetical ordering (AIM-ICF in Figure 4.54 and IKN-

SSR in Figure 4.55). CVS immediately stands out as being anomalously dominated by spatial considerations; it appears that the algorithm predominantly produces maps that predict fixation location as a function of distance to the centre of the image. Given this enormous apparent over-emphasis on central proximity, this model will be excluded from subsequent analysis noting patterns of NSS scores in relation to spatiotemporal considerations across algorithms.

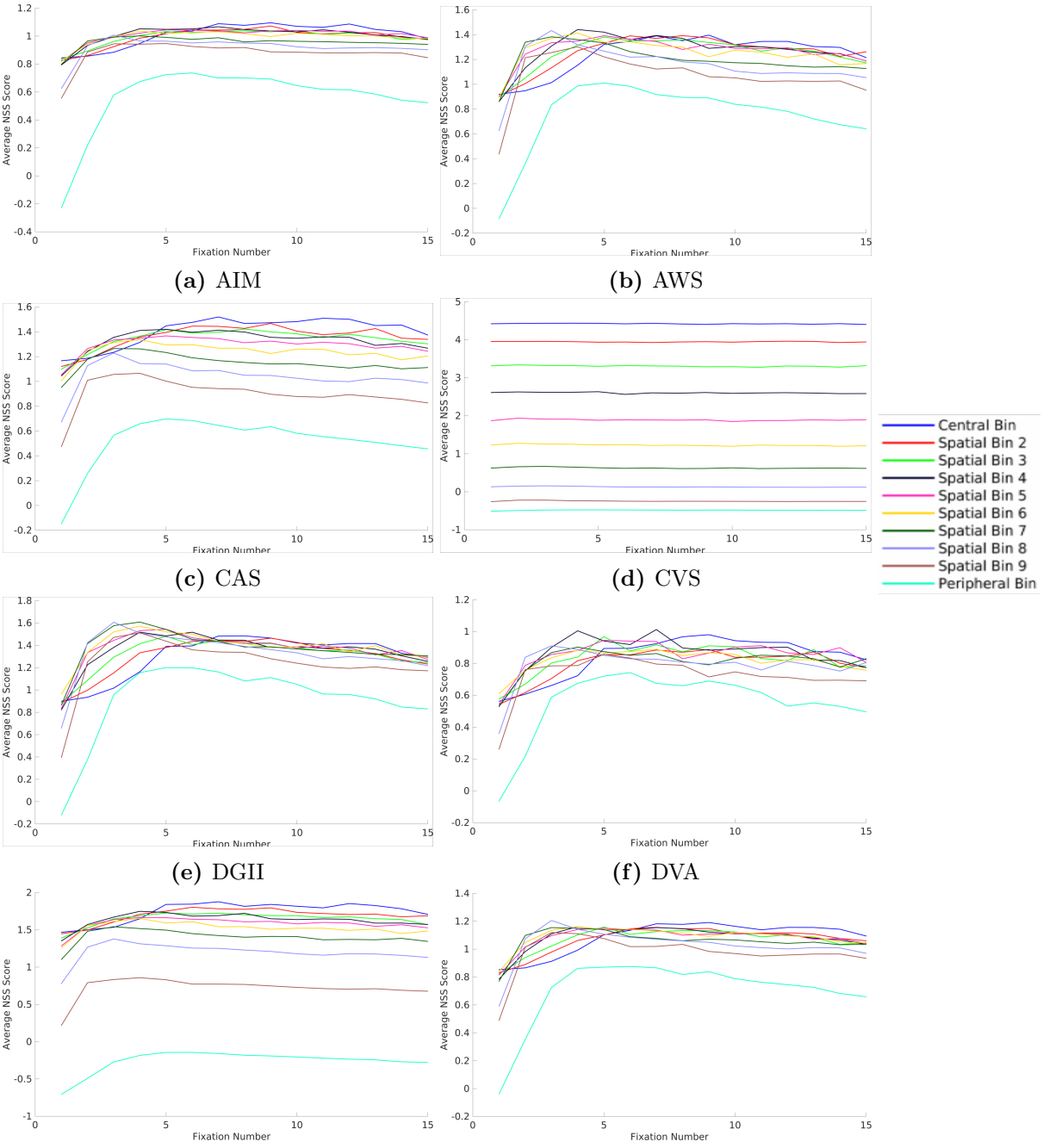
Although not exceedingly surprising in light of the results of Chapter 2, it is nevertheless surprising to see that the most peripheral bin shows substantially reduced NSS score compared to other spatial bins for nearly all models, even those (such as AIM and AWS) that were shown in Section 2.4 to be relatively free of spatial bias. QSS appears to be the only model that does not consistently report NSS scores in the most peripheral bin well below those of other spatial bins. SAM is unusual in that the two most central bins show a highly exaggerated increase in score, whereas for all other models (aside from CVS) the reduction in scores as one moves toward the periphery is much less marked.

It is interesting to note that the first fixation frequently has the *lowest* average score for a given spatial bin, a result that is completely counter to the expectations of prior research. While this appears to be consistent across spatial bins, the difference between first and second fixation scores is greater in more peripheral bins. Almost all algorithms exhibit a steady climb in central scores over the first five fixations, whereas more peripheral bins frequently peak in score during these early fixations and then subsequently decline. Aside from this initial uptick between the first and second fixations, many of the classic models (AIM, AWS, DVA, QSS, IMSIG, RARE2012, and SSR) display relatively stable scores over both time and space (discounting the aforementioned most peripheral bin), whereas for others with a larger spatial bias component (CAS, GBVS, and IKN) appear to predict the first several fixations relatively equally across most spatial bins but then settle into a distinct spatially organized spread (with higher average scores closer to the centre) that does not greatly vary temporally. This is not a hard separation; almost all of the algorithms appear to settle into at least a small spatial spread that favours more central fixations over more peripheral ones, but the magnitude of this separation is larger in CAS, GBVS, and IKN. Regardless of the spatial spread in scores, though, it is interesting to note that most classic algorithms show relatively stable scores across the temporal dimension after the first 1-5 fixations. A few models show a slight downward trajectory through time (most notably AWS, IMSIG, and QSS).

Most models that are based on deep learning (DGII, SalGAN, oSALICON, and SAM) show an interesting pattern that shows a marked peak in score for fixations located in the mid-range of spatial spread for the first several fixations, after which scores either become largely independent of spatial location with a slow temporal decline (DGII and oSALICON) or appear to settle into a pattern mostly dominated by spatial considerations (SalGAN and SAM). Perhaps most unexpected is the fact that for three of these models (DGII, SalGAN, and oSALICON) this 2-5 fixation period is marked by higher scores peripherally than centrally, after which the central scores climb and the peripheral scores drop. This is particularly unexpected for DGII, as it explicitly incorporates a central prior during post-processing. ICF is a somewhat unique model in that it is trained in a manner similar to other deep learning efforts, but is explicitly based on more classical features. ICF's behaviour appears in many ways to be somewhat intermediate between deep and classic models; it shows a similar but less exaggerated pattern in the 2-5 fixation range compared to DGII, SalGAN, and oSALICON, but settles into a subsequent pattern that appears somewhat closer to more classic algorithms such as AIM.

4.3.4.2 AUC Scores Over All Categories

Spatially binned AUC scores plotted across time are shown in Figures 4.56 (AIM-ICF) and 4.57 (IKN-SSR). As one can see, AUC scores are largely less spiky than NSS scores, and though the numerical values differ between algorithms the overall patterns are more consistent in the AUC plots. Nearly every algorithm finds that AUC for the most central fixations climbs through the first five fixations, whereas it usually peaks within this range for the more peripheral bins. For most models the scores for the central bins start near the bottom (with the common exception of the most peripheral bin), and then around the fourth or fifth fixation the rise in central scores and drop in peripheral scores leads to a period of time over which bins are scored in decreasing order from the centre to the periphery (some algorithms, such as AIM, DGII, and oSALICON have very low score differences between spatial bins after the fifth fixation). A few algorithms break this pattern slightly such that early fixations still exhibit a higher score in central than peripheral fixations (*e.g.* GBVS, SalGAN, and SAM), but even in these models this deviation from the aforementioned patterns appears to be the inherent central bias of these models, and there is still a characteristic rise in central bin scores over the first five fixations, whereas more peripheral bins peak and then



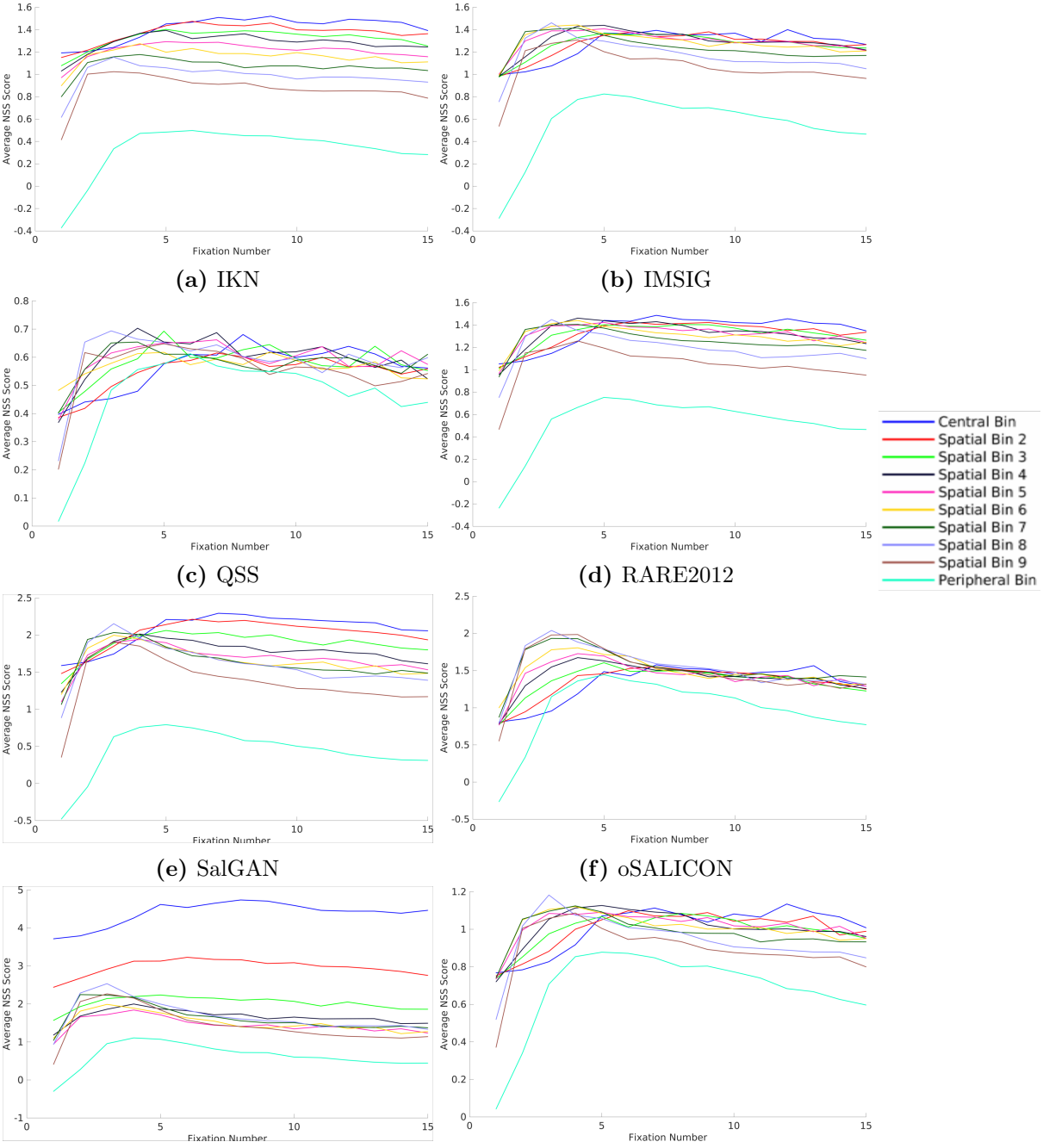
(a) AIM **(b)** AWS

(c) CAS **(d)** CVS

(e) DGII **(f)** DVA

(g) GBVS **(h)** ICF

Figure 4.54: Plots of NSS scores against fixation number for models AIM through ICF (alphabetically ordered). Scores are calculated individually for the spatial bins shown in Figure 4.50. Note that the range on the y-axis varies from model to model depending on the magnitude of NSS scores achieved by the model; the aim here is not to directly compare model scores, but rather to identify prominent spatiotemporal patterns that appear for each specific model.



(a) IKN **(b)** IMSIG

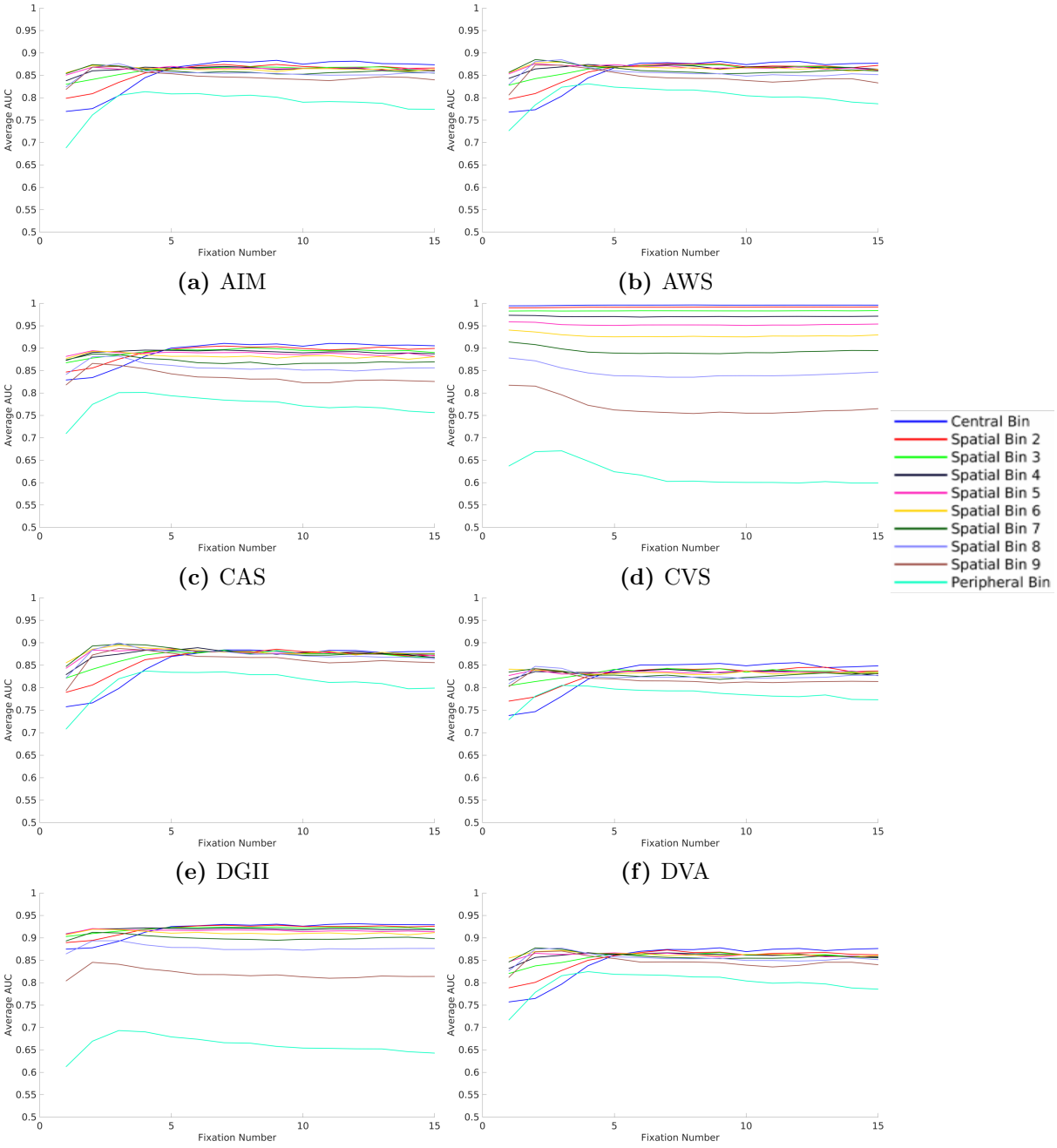
(c) QSS **(d)** RARE2012

(e) SalGAN **(f)** oSALICON

(g) SAM **(h)** SSR

Figure 4.55: Plots of NSS scores against fixation number for models IKN through SSR (alphabetically ordered). Scores are calculated individually for the spatial bins shown in Figure 4.50. Note that the range on the y-axis varies from model to model depending on the magnitude of NSS scores achieved by the model; the aim here is not to directly compare model scores, but rather to identify prominent spatiotemporal patterns that appear for each specific model.

drop.



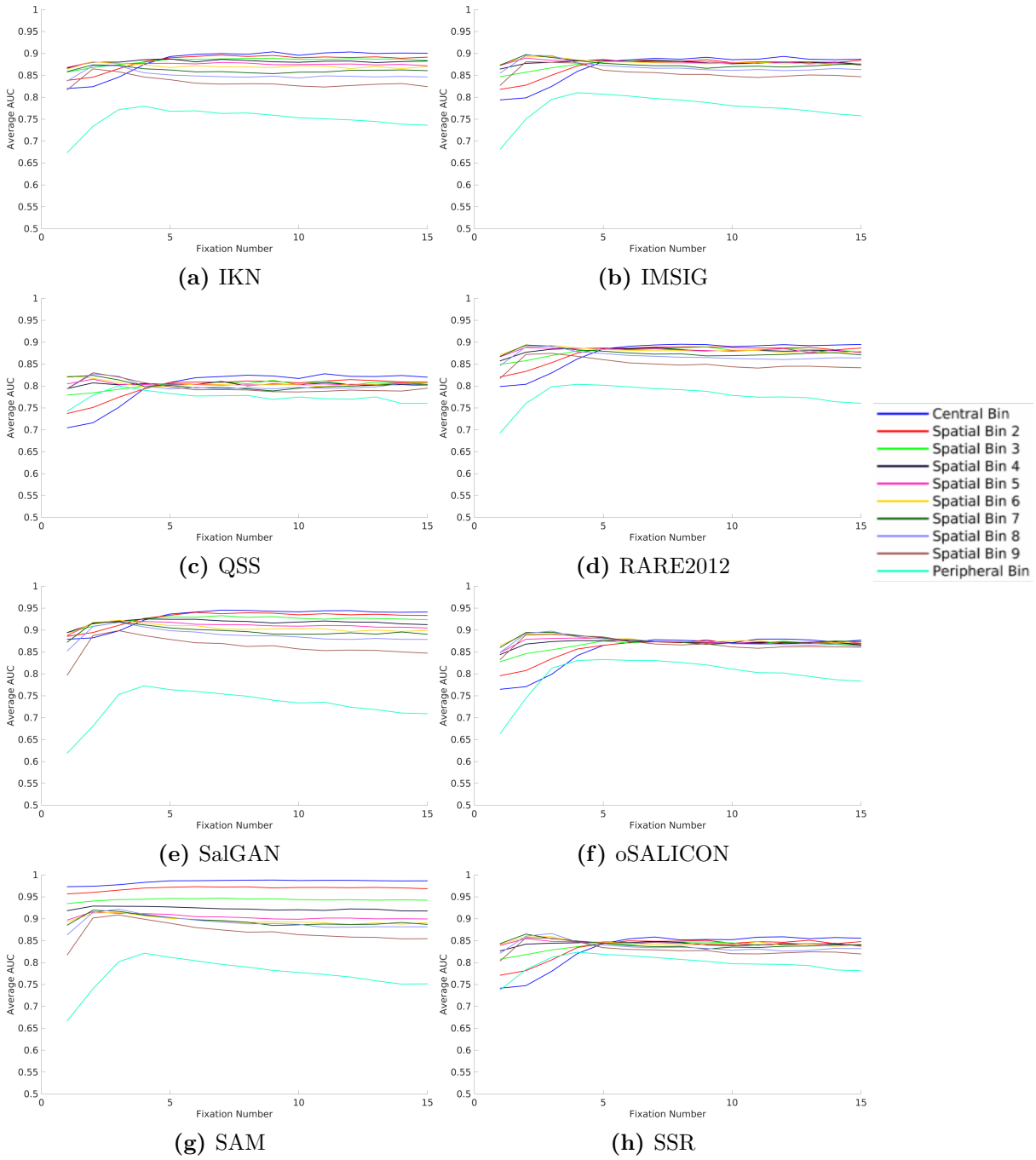
(a) AIM **(b)** AWS

(c) CAS **(d)** CVS

(e) DGII **(f)** DVA

(g) GBVS **(h)** ICF

Figure 4.56: Plots of AUC against fixation number for models AIM through ICF (alphabetically ordered). Scores are calculated individually for the spatial bins shown in Figure 4.50. Note that the aim here is not to directly compare model scores, but rather to identify prominent spatiotemporal patterns that appear for each specific model.



(a) IKN **(b)** IMSIG

(c) QSS **(d)** RARE2012

(e) SalGAN **(f)** oSALICON

(g) SAM **(h)** SSR

Figure 4.57: Plots of AUC against fixation number for models IKN through SSR (alphabetically ordered). Scores are calculated individually for the spatial bins shown in Figure 4.50. Note that the aim here is not to directly compare model scores, but rather to identify prominent spatiotemporal patterns that appear for each specific model.

4.3.5 Discussion

It is rather remarkable that for fifteen of the sixteen algorithms tested there are a couple clear and unexpected trends (with the outlying model, CVS, appearing to be overwhelmingly dominated by spatial bias). The first is that the first fixation is on average less salient than the subsequent four or five fixations, and frequently has the lowest score across all time steps tested. This pattern is more prominent in the NSS metric than AUC, but nevertheless appears in both analyses. This is a very surprising result that is not consistent with any of the prior literature on this topic (reviewed in Section 4.3.1); previously efforts have posited either that saliency provides a relatively constant contribution to fixation selection through time (*e.g.* [95]), or that early fixations are most strongly driven by saliency, with the first fixation in particular being the most strongly saliency driven (*e.g.* [96, 270]). The fact that the first fixation would exhibit the lowest overall score, regardless of whether a model is based on the sort of high-level features learned by deep networks or classical features motivated by psychology or information-theoretic principles, suggests a clear hole in our current understanding of saliency contributions along the temporal dimension.

The second notable pattern is that different spatial bins exhibit different temporal shifts in score, with more peripheral bins tending to achieve higher scores early (during the first 2-5 fixations) before their scores begin to drop or saturate, while central bins tend to rise steadily through these first several fixations before overtaking the scores of the more peripheral bins and saturating at a higher value. There are some differences across models, such as those with a stronger centre bias (*e.g.* GBVS) shift central score curves up, but the shapes of the curves (rising early for peripheral bins, and more slowly for central bins) still appears to be relatively consistent.

Likely a strong factor into both of these patterns is the penchant for human observers to rely on information sampling strategies posited by visibility models (see Section 1.2.2.6). In particular, Tatler [193] demonstrated that when observers' initial fixations are not in the centre of an image they will very frequently fixate the centre with their first saccade, regardless of the specific image content. Tatler hypothesized that this was based on the centre being a good sampling point for getting an overview of the image. It could therefore be that when earlier fixations are directed away from the image centre they are more likely to be driven by the attraction of a specific target, whereas more central fixations may be drawn by a target but could also be selected due to the

strategic value of that location (thereby reducing the average correlation of central fixations with the underlying saliency representation).

Another factor which should be acknowledged, particularly related to the low peripheral scores for the first fixation, is that the saccadic targeting is subject to noise or spurious eye movements. As shown in Figure 4.51, early fixations are heavily skewed towards the centre, meaning that the number of fixations falling in more peripheral spatial bins is much lower. Any spurious fixation (which is unlikely to correlate highly with a saliency map) is likely to much more heavily impact the calculated score for these early bins than it would a bin with a larger number of recorded fixations.

This experiment provides strong evidence that there are subtle but consistent spatiotemporal patterns to the contribution of saliency to fixation selection. The consistency of these effects across models points toward them being an attribute not of any one particular formulation of saliency, but rather as a systemic factor in human patterns of observation. Extending fixation prediction models to explicitly incorporate temporal aspects, such as the model presented in Chapter 5 should allow us to improve our understanding of human fixation selection. Future extensions ought to explore the merging of saliency representations with the higher level viewing strategies found in visibility models.

4.4 Conclusions

This chapter has presented three different experiments exploring the patterns and behaviours of a broad spectrum of saliency models. This work is complementary to standard performance benchmarking efforts, and provides new insights and highlights potential challenges for saliency modelling that might otherwise be overlooked in the pursuit of benchmarked rankings over a narrow field of established test sets composed predominantly of natural images.

While each study naturally requires effort to develop a clear and well-framed scientific question, SMILER minimizes a significant practical impediment to this sort of research by greatly reducing the effort needed to produce a set of saliency maps from a diverse selection of models. Additionally, the work presented here can much more easily be independently replicated or extended due to the ease of use provided by SMILER. It is hoped that SMILER will enable more research like this in the future in order to better develop the theoretical foundations and implications of saliency modelling

research.

Chapter 5

Explicit Fixation Control

Portions of the work in this chapter have been published previously as the following:

John K. Tsotsos, Iuliia Kotseruba, and Calden Wloka, "A Focus on Selection for Fixation", *Journal of Eye Movement Research*, vol. 9, pp. 1-34, 2016

In this work John was the primary author, with editorial feedback from Iuliia and Calden. Calden provided 90% *of the coding for the fixation controller implementation described, and executed the qualitative experiments. Iuliia provided the other* 10% *of the coding for the fixation controller model, as well as the analysis covered in "The Role of Saliency Maps" section (this work is not included here). The qualitative experiments described in this paper are presented in Section 5.5.1, followed by a set of novel extensions presented in Section 5.5.2.*

Calden Wloka, Iuliia Kotseruba, and John K. Tsotsos, "Active Fixation Control to Predict Saccade Sequences", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June, 2018

In this work Calden was the primary author, with editorial feedback from Iuliia and John. Iuliia is responsible for 60% *of the coding implementation used in this work; the original model code was implemented by Calden, and subsequently re-implemented by Iuliia with a focus on improvements to its computational efficiency and stability. The quantitative results presented in this paper are included here in Sections 5.6 and 5.7. All experiments and demonstrations included in this paper were collaboratively conducted, with approximately* 30% *of their design determined by Iuliia and the other* 70% *determined by Calden. Calden is primarily responsible for the theoretical design of the model, and interpretation and analysis of the results.*

5.1 Motivation

Most applications in computer vision function primarily in a passive way; algorithms are applied to static images or pre-recorded video sequences without control over what visual data is acquired next. However, it has long been recognized that eye movements are an integral aspect to human vision [40], with diverse functionality ranging from the enhanced extraction of features via microsaccadic motion [275] through high-level strategies for optimal information gathering [115]. It is this latter aspect that is of particular interest to the field of computer vision; active control over the acquisition of image data is fundamental to efficiently developing more robust and general computer vision solutions for unconstrained environments [276, 277].

The topic of active control has a natural connection to the topic of saliency modelling. Eye movement control and early visual attention have frequently been conflated, particularly within the computational saliency literature. As mentioned in 1.2, the highly influential Itti-Koch-Niebur (IKN) model [25], which grew out of theories of attentional gating and pre-attentive processing found in the work of Koch and Ullman [20] and Treisman and Gelade [21], popularized and entrenched the idea of a static heat map representation for ordered attentional processing in an image. However, the significant challenge of obtaining a suitable source of ground-truth data with which to validate a map of pre-attentive processing led to a shift in focus to predicting fixation locations. Though this shift in focus provided a readily available source of data in order to establish performance benchmarks, it was an unspoken and never justified switch from the modelling of exogenous attentional capture (whether overt or covert) to a sole focus on predicting overt shifts. Though this change in direction of the field has important ramifications for the intellectual stability of the field of saliency modelling, the fact that overt shifts in fixation represent a primary component of active vision in biological systems allows a number of aspects of saliency modelling research to provide potential insight into the field of active vision.

Of course, *active* implies a dynamic component. While a large number of saliency algorithms have been developed over the years (see Section 1.2.5 for an overview of a wide ranging selection), the emphasis of fixation prediction has largely rested on the development of static heat maps over an image. Evaluation metrics (reviewed in Section 2.2) likewise treat fixation prediction largely as a purely visual task, treating the fixation targets of all subjects in the same equivalent pool of

locations regardless of temporal sequence or spatial layout. While this focus has come a long way in helping to identify the visual elements that contribute to attracting fixations during free viewing, it is missing important components to the control of gaze (for example, in Chapter 2 it is argued that spatial layout is not independent of visual features, and therefore the spatiotemporal nature of fixation sequences is a primary source of the centre bias seen in fixation datasets over natural images).

This chapter attempts to move beyond the static treatment of saliency prediction, and presents the Selective Tuning Attentive Reference Fixation Controller (STAR-FC) model. STAR-FC serves as an explicit model of human saccadic control. In order to more easily compare with prior efforts in fixation prediction, we concentrate on the free-viewing paradigm, but nevertheless specify our control network in a manner that provides explicit extensibility for task-based tuning and top-down attentional control. By providing an extensible model of human fixation control that includes a number of aspects normally neglected by the saliency literature, including an explicit transform to account for anisotropic retinal acuity, we are able to produce explicit fixation sequences with greater fidelity to those of humans than is seen by traditional saliency approaches (see Figures 5.16-5.18 for qualitative examples, and Section 5.6 for a quantitative comparison to human and saliency map performance). STAR-FC provides a platform not only to better understand and incorporate the principles of active vision into machine vision applications, but also to explore the function of early human attention.

It should be noted that outside of the saliency literature there are a number of eye movement control models. However, such models are usually dedicated to a specific subset of eye movements (such as smooth pursuit [278], the optokinetic reflex [279], or 3D gaze shifts [280]) or neural component (such as the role of the superior colliculus [281], cerebellum [282] or the basal ganglia [283]) without a clear path of extension or inclusion of attentional control. As much as possible, the insight into eye movement control provided by these models has informed the development of STAR-FC such that it should remain extensible in the future to incorporate these additional aspects of gaze control without necessitating a large architectural shift.

5.1.1 Applications of Fixation Prediction

Early interest in saccadic sequences was heavily influenced by Noton and Stark's *scanpath theory* [284], which posited that the explicit spatiotemporal structure of eye movements drove memory encoding for visual patterns and subsequent retrieval. However, challenges to this view have arisen over the years, among them experimental evidence showing that there is no recognition advantage conferred by the use of one's own fixation locations versus those of another viewer nor by the retention of the temporal order of fixation [285]. These results certainly support the traditional approach to saliency evaluation that predominantly seeks to evaluate algorithms on prediction effectiveness over a static ground-truth fixation cloud, disregarding individual source and temporal characteristics of the fixations.

However, scanpath theory was largely devoted to explaining memory encoding and the recall of images. Even if visual memory is not heavily influenced by scanpaths, there are nevertheless a number of applications for which explicit fixation sequence modelling and prediction is very valuable. For example, motivated by the very short window of consumer attention to most advertisements, commercial applications of saliency analysis already include predicted sequences of the first several fixations [286], despite validation using only traditional ROC methods that do not measure the efficacy of sequence prediction [287].

Similar to applications in commercial design, there has been recent interest in understanding eye movements over scientific figures in order to promote better communication of vital scientific information to both the public and to policy makers. Harold *et al.* [288] specifically focus on the topic of climate change communication, and use data acquired from eye tracking to motivate their research. However, their findings are not limited solely to the field of climate change, but are rather important to all areas of science with impact on public policy. Even more so than in advertising, the sequence of fixations becomes important for understanding whether and how viewers are understanding scientific figures, as a correct interpretation of one part of a figure may be dependent on information having already been attended to in another location. It would be exceedingly expensive to perform eye tracking for even a subset of scientific figures, but an automated sequential fixation prediction system could potentially provide great benefit without the prohibitive cost and logistical challenge of human eye tracking.

As previously mentioned, understanding the control of human eye movements may additionally be highly instructive in robotic visual systems with active camera control, as in [22, 30]. This is particularly useful for applications with anisotropic sensors that could be considered analogous to the anisotropy present within the human retina, such as omnidirectional camera systems that introduce a high degree of spatial distortion unevenly across the visual field [289] or a two-camera visual input system that combines high- and low-resolution streams to effectively maintain a wide field of view without sacrificing the ability to acquire high acuity detail over a targeted region [290]. Furthermore, as robotic applications increase their focus on social interactions, it becomes important not only to accurately attend to relevant information during an interaction, but also to provide socially important cues through body language such as gaze location [291]. Robotic modelling of joint attention has previously been improved through the application of saliency [292], and can likely be further improved with a more complete gaze model. Accurate modelling of joint attention between parties has wide reaching ramifications, from self-driving vehicles [293] to the handover of physical objects [294].

5.2 Saccade Sequence Generation over Static Saliency Maps

While our goal differs from the standard manner in which saliency algorithms are applied and evaluated, the calculation of static maps nevertheless represents the dominant computational model for fixation prediction and will therefore serve as the primary point of comparison. Static saliency maps have previously been used to generate explicit fixation sequences directly, such as Itti and Koch's [64] search system that couples winner-take-all selection to a simple inhibition of return scheme. Likewise, as reviewed in Section 1.2.5.4, a number of methods have approached the connection between explicit eye movement patterns and saliency maps from a different direction by attempting to learn a motoric prior from human fixation data, and then applying that information over a static underlying map.

Most such sequence prediction models operate via a stochastic process that evolves over an underlying static saliency map. Brockman and Geisel [203] provided an early example by proposing that gaze shifts could be modeled as a Lévy flight random walk [295] over an attractor field (the saliency map). While this provided a much more qualitative appearance of fixation locations, it

was unclear how to tune, train, or otherwise modulate their method. Tatler and Vincent [170] developed a saliency algorithm that was independent of the visual input and instead entirely based on statistical regularities in eye movements. Despite the lack of visual processing, they nevertheless demonstrated comparable or better performance than the IKN saliency model, suggesting that fixation location may be driven as much by the underlying motor control of the eye as it is by visual information. They likewise found that by combining their motor model with the IKN saliency map they could achieve performances that outperformed each independent component.

Mathe and Sminchisescu [296] approached the topic of fixation sequence prediction under task-specific conditions (action and context recognition). They define a model to automatically recover Areas of Interest (AOIs) from human data and then train a Hidden Markov Model (HMM) to calculate transition probabilities between AOIs. Inverse reinforcement learning is then used to train a method for predicting sequences over novel images. It is interesting to note that much of the performance of this work seems driven by the task constraint; as mentioned in Section 1.2.3, imposing semantic structure (whether through the inherent narrative flow of a video or the task guidance introduced by Mathe and Sminchisescu) serves to increase the correlation of fixations between individual subjects, and it is not clear if free viewing conditions would provide sufficient information for their model to extract useful AOIs. Liu *et al.* [297] use a similar HMM approach, but focus on the task of free-viewing. A Bag of Words model over visual features is trained to provide candidate regions, and then an HMM learns the transition probabilities between these regions.

More recently, Le Meur and Liu [11] provide an updated effort in this vein, training a stochastic Markov process for saccade transitions that, along with a memory state, modulates the underlying saliency map to produce a distribution over the next likely fixation point. The model may be used in either a deterministic WTA fashion to sample a maximum-likelihood sequence, or stochastically to model a number of observers with the degree of inter-observer variability that is parametrically controlled. The purpose of this model, however, is nevertheless to refine a final static output map, and its formulation does not lend itself to the generation of an explicit sequence prediction.

Jiang *et al.* [298] take a slightly different approach to the topic, learning a visual exploration policy using least-squares policy iteration rather than explicitly attempting to learn the transition probabilities in a Markov model. This approach is somewhat more flexible and semantically in-

terpretable than the Markov model-based approaches, and allows for an interesting exploration of low-level vs. high-level gaze cues as well as temporal epoch splitting.

Most recently, several purely deep learning approaches have emerged in this area. Ngo and Manjunath [299] provide one of the earliest such models using an LSTM module to provide a temporal component. However, the code for this method was not available at the time that the work in this Chapter was performed. Subsequent to the original publication of the work presented here, several additional deep learning based models predicing fixation sequences have been released, including the IOR-ROI LSTM model [300] and PathGAN model [301].

While these efforts do attempt to more carefully model the spatiotemporal aspects of saccadic control than most static fixation prediction saliency models, there remains an important difference between their methodology and STAR-FC. Although the research that goes into informing the motoric bias prior for the stochastic process do provide some interesting and valuable insights into saccadic control (see, for example, [171] for a very interesting dicussion of saccadic distributions), the models themselves are primarily learned performance models. As was shown in Section 2.1.1, even within the MIT dataset there is a distinct shift in the saccadic amplitude distribution between portrait and landscape images, suggesting that there may not be a single distributional prior that will serve. Any sufficiently large change in the experimental protocol under which fixations are obtained would likely necessitate an additional learning stage and access to sufficient data to retrain priors. While Le Meur and Coutrot [171] do show that there may be contextually learned spatial biases that likely cannot be mechanistically modeled intrinsicly within a saccadic control system (for example, the very distinct saccade distribution seen over webpages is most likely a product of our expectation for how information is laid out and the direction in which we read), the goal of STAR-FC is to provide a mechanistic explanation for as much as possible. The spatiotemporal properties of saccadic vision, such as retinal anisotropy, are relied on directly to modulate the spatial distribution of saccades, while the model itself remains extensible to any non-mechanistic priors that may be added in the future.

Finally, it is important to note the work of Sun *et al.* [302], who defined saliency using Super Gaussian Components. Fixations were generated by calculating maximum projections through this feature space, while orthogonality constraints ensured that subsequent fixations would not always converge on the same location (effectively serving as a form of IOR). Although their method does

provide explicit sequences, the sampling method does not generalize to other definitions of saliency, and our comparisons therefore focus on the more common approach of WTA sampling from static maps.

5.3 Visibility Models and STAR-FC

As discussed in Section 1.2.2.6, there is a robust body of work that aims to explain saccadic targeting more from an information gathering strategy perspective than from the salient pull of visual stimuli. Neither visibility models nor purely stimulus-driven models fully capture the extent of human behaviour, and thus an outstanding research goal in the field is to merge the two domains of thought. While STAR-FC as presented here is formulated as stimulus-driven saccade targeting, the larger STAR model [98] includes task and executive components. Therefore, a natural avenue for future development will be to incorporate the strategic, information gathering considerations of visibility models to the stimulus-driven conspicuity drive that is already well captured by STAR-FC.

5.4 Model Description

The STAR-FC model is designed to provide a comprehensive and biologically plausible model of saccadic eye movements in a manner that retains clear avenues for future extension and incorporation into a larger model of visual cognition and eye control. The conceptual formulation of the model is shown in Figure 5.1, while a schematic showing the implementation used for the experiments described later in the chapter (with the exception of Section 5.5.1) is shown in Figure 5.2.

5.4.1 Theoretical Formulation of STAR-FC

Figure 5.1 provides an abstracted representation of the STAR-FC model. Trapezoidal boxes denote neuronal representations of information, whereas rounded rectangles denote a process that is executed over the information provided to it. Arrows represent non-linear weighted sum connections between these representations and processes.

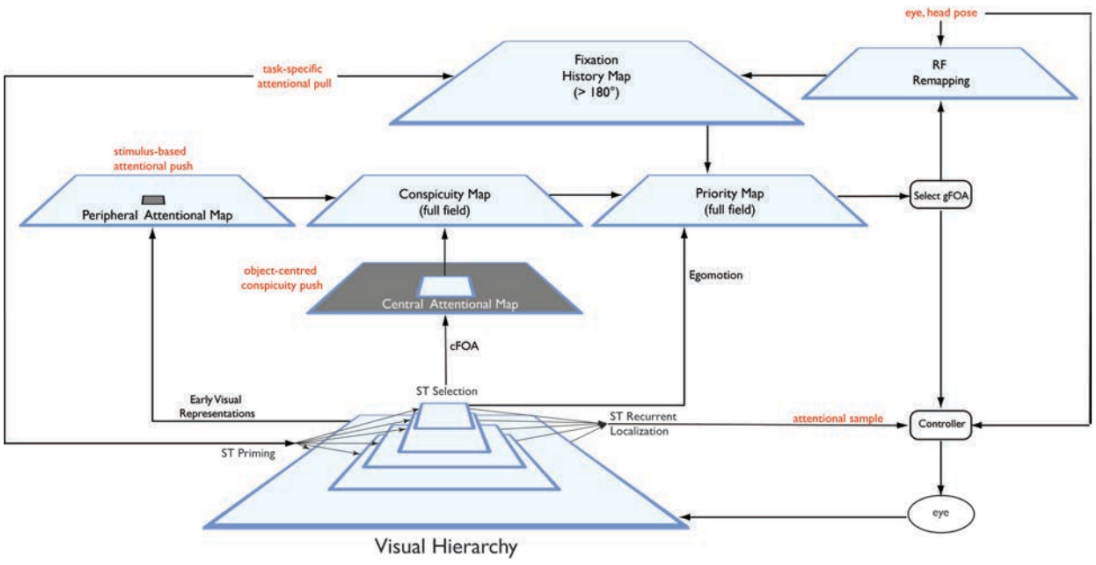


Figure 5.1: The conceptual framework for the STAR-FC model. Component descriptions are given in the main text.

Each iteration over the network begins with the acquisition of visual information from the eye, which then projects information to the visual hierarchy. One of the prominent assumptions made in the formulation of the STAR-FC model is that there is no early-gating function, and instead any mechanisms for saliency assessment are based on the feature processing already being calculated within the visual hierarchy. The role of saliency itself is to serve as a component in the decision process for the allocation of attention. This need not be through overt eye movements, though for the free-viewing task the assumption is generally made that explicit eye movements are made in conjunction with shifts of attention, and the current implementation of STAR-FC therefore also initiates an eye movement with each selection of a new focus of attention. It is important to note that this is a matter of convenience for comparison against human fixation trajectories; as discussed in Section 1.2.1 attentional capture does not necessarily require an overt shift in attention. The theoretical model for STAR-FC encompasses covert and overt shifts as required for a given task, though this portion of the model has not yet been implemented or tested.

The visual hierarchy itself may be modulated by *priming*, which is the process by which task-specific or prior knowledge modulates the processing and interpretation of visual features (see [19] for a discussion of the top-down biasing of information through Selective Tuning, and [303] for a proposed mechanism of neural network priming). The full hierarchical representation of the

attended subject forms the *attentional sample*, which is a binding of feature activations across the different layers of abstraction within the visual hierarchy (see [19] and [48] for a more extensive discussion of feature binding in attention). The attentional sample is sent to the executive controller of the network, as the information extracted may inform future iterations through the network (not shown in the diagram is a working memory component, but future extensions that include memory would likely interact with the fixation control system by receiving and maintaining information extracted in the form of this attentional sample).

As the visual hierarchy processes input, the activity of its neurons is collected into two streams of stimulus-driven attentional push: the *Peripheral Attentional Map* and the *Central Attentional Map*. The peripheral representation may or may not extend into the central region (see Section 5.4.2 for a discussion of different merging strategies), but the central field represents information at the highest levels of visual abstraction that is therefore restricted by the Boundary Problem of hierarchical processing (see [19] for a general discussion of the Boundary Problem in visual processing, and [97] for a discussion of the specific relationship between the Boundary Problem and fixation control). The issue of an undefined convolutional boundary has been acknowledged previously as a source of implicit centre bias in saliency models (see Section 2.1.1), but remains a concern, particularly as neural network depth increases. By introducing two streams of processing along with a moving centre of gaze, these issues can be greatly reduced when operating in a real environment. Likewise, having two streams of stimulus-driven attentional push also allows for a rather succinct solution to the debate between whether saliency should be object-based or feature-based raised in Section 1.2.2.1: why not both? By accessing responses from the full extent of the visual hierarchy, the central field provides all the object-level information available from a feed-forward pass of the visual hierarchy, allowing the primary driver of activity in the central attentional field to be object-based. The peripheral attentional field, in contrast, is driven by activity in only the earlier layers of the visual hierarchy (before the Boundary Problem becomes an extensive issue), and thus represents a more feature-driven representation of saliency.

The two streams of processing are re-integrated in the *Conspicuity Map*, which provides a full-field representation of the stimulus-driven attentional push. This integration is non-trivial, as both higher-level abstraction and low-level features appear to contribute importantly to fixation selection [184]. It is therefore necessary for the range of possible output values from each stream

to be able to occasionally override the other and drive selection. Nevertheless, while the current model handles this integration with a static method (see Section 5.4.2 for details on the integration strategies tested), the integration of central and peripheral streams for stimulus-driven attentional push provides a potential avenue of extension for incorporating additional modulatory mechanisms in saccade control (such as affective influences [304], which are currently not well-reflected within the saliency literature [305, 164]).

The *Priority Map* follows in the integrative capacity proposed by Fecteau and Munoz [59]; it serves as the final representational stage before a stimulus-driven global Focus of Attention (gFOA) is selected via a WTA process. In addition to the input from the Conspicuity Map, the Priority Map is also modulated by the Fixation History Map (described below) and egomotion cues (such as those generated via pursuit movements).

The *Fixation History Map* provides spatial biasing and modulation for the priority map. This includes Inhibition of Return (IOR), which reduces the drive to fixate previously fixated targets or locations unless there happens to be a task-specific reason to do so [306], as well as any top-down spatial priors (collectively referred to as task-specific attentional pull). These top-down priors could include biasing based on scene type (such as the expected layout of information for a website [171]) or task (such as the tendancy to view the road directly ahead or the mirrors while driving). The Fixation History Map's representation of space is necessarily larger than the visual field, as it must maintain information even when it goes out of view. *RF Remapping* provides the neural computation to keep the Fixation History Map updated in a gaze-centered coordinate system by translating the stored information each time the eyes move.

5.4.2 Implementation Details of STAR-FC

Figure 5.2 presents the currently implemented formulation of STAR-FC. Although it conceptually follows the theoretical framework described in Section 5.4.1, the scope of implementation necessitated that some aspects of the theoretical model remain open for future work. Of particular note is the lack of a full implementation of the visual hierarchy consistent with the Selective Tuning theory [18, 19] as well as the corresponding executive control modules. Without the task biasing both qualitative (Section 5.5) and quantitative (Section 5.6) tests of STAR-FC's function concentrate on the free-viewing paradigm.

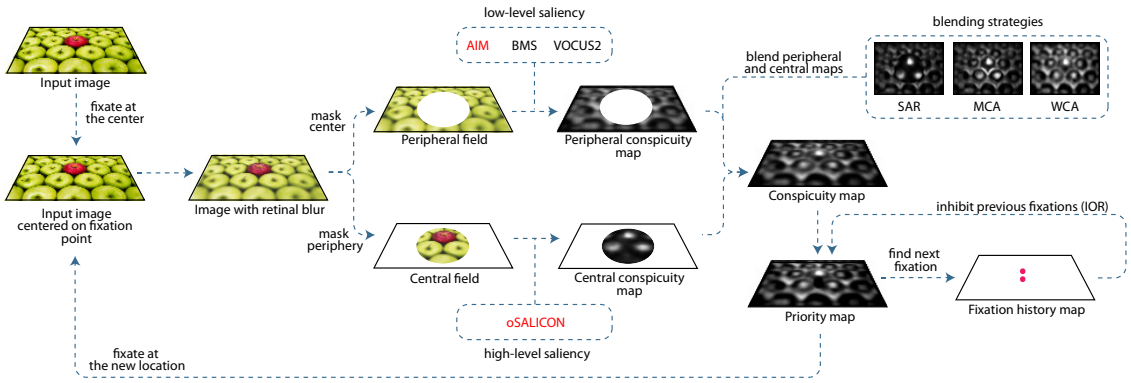


Figure 5.2: A schematic of the STAR-FC implementation.

Without a full implementation of the visual hierarchy, the retinally transformed image is passed directly to the saliency models that provide the central and peripheral stimulus-driven attentional push streams. For more details regarding the retinal transformation employed by STAR-FC, see Appendix B.1. Low-level models that operate over only a small number of processing steps are used in the periphery, and deep models that better approximate a full pass through the visual hierarchy serve in the central stream.

A number of different peripheral field models were experimented with, including the BMS model [307], the VOCUS2 model [150], and several bases sets for the AIM model [2]. For specific details regarding model calculations, see the corresponding model overviews in Section 1.2.5. Overall it was found that AIM using the 21infomax950 basis provided the best performance, and therefore all results reported in this chapter are obtained using this method in the peripheral field. Likewise, with the exception of Section 5.5.1, all results reported in this chapter are obtained using the OpenSALICON [220] implementation of the SALICON saliency model [181] in the central field. Further details regarding the performance of the different peripheral field options can be found in Appendix B.

Three strategies were implemented to blend the peripheral and central streams: Separate Activation Regions (SAR), Maximum Central Activation (MCA), and Weighted Central Activation (WCA). For all three strategies, the following notation is used: CM_{ij}, C_{ij}, and P_{ij} are the conspicuity map, central map, and peripheral map values, respectively, at pixel (i, j). r_{ij} is the distance from the image centre for pixel (i, j), and r_c refers to the radius of the central field. g_p represents an optional peripheral gain factor that may be used to increase the importance of peripheral features

most affected by the retinal transform.

- *Separate Activation Regions* (SAR): A binary mask is applied to both the peripheral and central attentional maps to confine activations to their respective fields. A narrow overlap region is included within which the maximum value of either the peripheral or central activation is retained (originally proposed in [97]). Thus, conspicuity map values are calculated as follows:

$$CM_{ij} = \begin{cases} C_{ij}, & \text{if } r_{ij} < r_c \\ \max(C_{ij}, g_p P_{ij}), & r_c \leq r_{ij} < r_c + \delta \\ g_p P_{ij}, & \text{otherwise} \end{cases} \tag{5.1}$$

  where δ is a small value specifying the thickness of the overlap region.

- *Maximum Central Activation* (MCA): The central attentional map is masked as in SAR, but no mask is applied to the peripheral map. The central region of the conspicuity map is equal to the maximum activation of either the peripheral or central maps according to the equation:

$$CM_{ij} = \begin{cases} \max(C_{ij}, P_{ij}), & \text{if } r_{ij} < r_c \\ g_p P_{ij}, & \text{otherwise} \end{cases} \tag{5.2}$$

- *Weighted Central Activation* (WCA): The peripheral and central attentional maps are combined as follows:

$$CM_{ij} = \begin{cases} \frac{r_c - r_{ij}}{r_c}\ C_{ij} + \frac{r_{ij}}{r_c} Pij, & \text{if } r_{ij} < r_c \\ (1 + \frac{r_{ij} - r_c}{r_{\max} - r_c})[1 - g_p] P_{ij}, & \text{otherwise} \end{cases} \tag{5.3}$$

  where $r_{\max}$ is the maximum distance from the centre for a given image.

The extent of the central field is set to 12.5 degrees. This decision is based on the fact that it roughly corresponds to the half-width of the average-sized receptive field in the inferotemporal (IT) cortex [308], which sits near the top of the ventral processing stream [46] and almost completely lacks receptive fields that do not overlap the fovea or parafovea [308].

The FHM is populated and operates in the following manner: When a new fixation is made, all previous elements of the FHM must be spatially updated to maintain a gaze-centered coordinate

frame. The FHM values also undergo linear decay. Thus, for a saccadic displacement of (x, y), the value of the (i, j)th pixel, FHM_{ij}, is updated according to the equation:

$$FHM'_{ij} = \max(0, FHM_{i+x,j+y} - \frac{1}{D_r}) \tag{5.4}$$

where D_r is the decay rate specifying the number of fixations which must elapse before the full IOR effects disappear. After updating the previous values of the FHM, a new IOR blob must be added to the map for the current fixation. For all pixels within the IOR radius r_I of the current fixation point, the pixel value of the (i, j)th pixel, FHM_{ij}, is updated according to the equation:

$$FHM'_{ij} = \min(1, FHM_{ij} + (1 - \frac{d_{ij}}{r_I})) \tag{5.5}$$

where d_{ij} is the distance of the (i, j)th pixel from the fixation point.

In the current formulation of STAR-FC FHM values are always inhibitory, and thus the values of the (i, j)th pixel of the Priority Map, PM_{ij} is equal to the equation:

$$PM_{ij} = CM_{ij} - FHM_{ij} \tag{5.6}$$

Once the Priority Map has been calculated, a new fixation is selected by selecting the pixel with the maximum value in the map.

5.5 Qualitative Demonstrations: Comparisons with Yarbus Traces

The field of eye tracking was pioneered by the seminal work of Yarbus [10], which offers a natural starting point to test a fixation control model. Yarbus' results differ from most modern eye tracking data in a number of important ways. Due to the manner in which Yarbus recorded eye traces (projection onto a photosensitive panel), his recordings were analog and without precise coordinates; it is therefore not possible to quantitatively compare the performance of STAR-FC against the human traces Yarbus recorded. Additionally, most modern eye tracking experiments that are used to test saliency model performance have rather short viewing durations (typically less than thirty seconds). Yarbus, by contrast, experimented with a number of different recording durations, often

asking subjects to view an image for several minutes. This creates a much longer exploratory trace than is typical in modern experimental data, which makes Yarbus' results an interesting qualitative comparison for STAR-FC.

As will be discussed in further detail in 5.6, Tatler *et al.* [95] and the analysis presented in Section 4.3.3 have previously shown that early fixations tend to be the most correlated between subjects, and subsequent quantitative analysis will therefore be restricted to the first five fixations. Nevertheless, even if the specific fixation locations in longer sequences are not being driven solely by saliency (and thus may deviate in terms of specific targets), the qualitative behaviour of the fixation patterns may provide a useful attribute for comparison. As will be seen below, static saliency maps have a hard time reproducing long sequences that resemble human patterns of fixation, even if many of the target locations are in common with the fixation targets of human observers.

5.5.1 Face Templates for the Unexpected Visitor

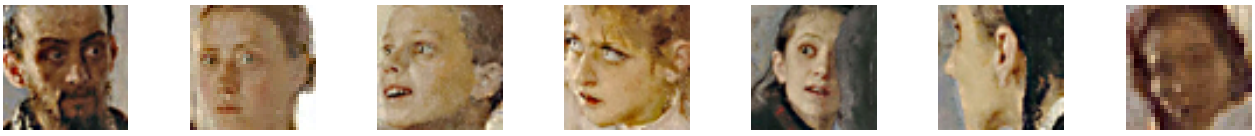
Ilya Repin's *Unexpected Visitor* (shown in Figure 5.3) is featured prominently in Yarbus' work [10], and provides a natural starting point for a proof of principle test of the STAR-FC architecture. At the time of the earliest implementation of the STAR-FC architecture it was unclear what the best form of central field attention would be. Although a number of deep learning models showed promising results on commonly used eye tracking datasets (see Section 1.2.5 for more details) and were analogous in depth to a feedforward pass of the visual hierarchy (making them a natural choice for the central field saliency model in the STAR-FC architecture), code was not yet available for any of these algorithms. It was therefore decided that an initial qualitative test of the STAR-FC architecture would be performed on the *Unexpected Visitor* using a central field that concentrated on faces.

Two aspects of the test over the Unexpected Visitor made the use of a standard face detection algorithm (such as the Viola-Jones face detector [173]) particularly challenging: The Unexpected Visitor is a painting rather than a natural image, and the STAR-FC architecture includes a retinal transform step. Even without the retinal transform it was found that numerous available face detectors failed to detect several of the faces at any reasonable detection threshold that did not introduce dozens of false positives. Once blurring from the retinal transform was added in the issue was only exacerbated. It was therefore decided that, given that the emphasis of this test

Figure 5.3: Ilya Repin's *Unexpected Visitor* painting.

was not on face detection but rather on the ability of the collective parts of STAR-FC to function together to create more human-like sequences than from a static saliency map, a template search using the target faces from the image itself (shown in Figure 5.5c) provided the most expedient way of performing the test. Using templates from the target image, while clearly lacking in generality and robustness, has the dual benefit of ignoring the issue that the image is a painting rather than a natural image and allows detection scores to degrade gracefully under distance from the fovea, rather than the sharp dropoff as was seen with face detection algorithms. This approach is similar to experimental tests of Zelinsky's Target Acquisition Model [106], except in this case there are multiple target templates and search does not end when any one of them is located.

In addition to the long viewing time, one advantage of using the *Unexpected Visitor* painting is that Yarbus performed both free-viewing and task-guided recordings of human eye movements. Of particular interest for this comparison is the sequence recorded when the viewer was instructed to estimate the age of the people in the painting, as this task heavily emphasizes the appearance of a person's face. As with the free-viewing traces (viewing the painting without any specific instructions or task) examined in Section 5.5.2 and shown in Figure 5.6, the trace shown in Figure 5.5b was recorded over a three minute interval.



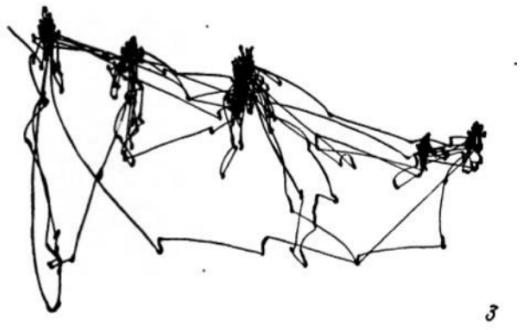
Face 1 Face 2 Face 3 Face 4 Face 5 Face 6 Face 7

Figure 5.4: The set of seven face templates used to produce the results shown in Figure 5.5c

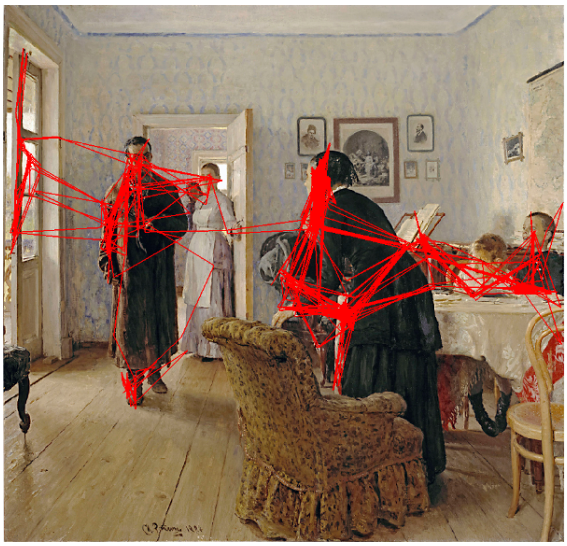
Assuming an average of four saccades per second[1], three minutes of viewing time works out to 720 fixation points. The results of generating 720 fixations using STAR-FC with a face template-based central field are shown in Figure 5.5c. As can be seen, this test produced an encouraging result with a qualitatively similar sequence to the human traces shown in Figure 5.5b. A wide search through parameter space was not performed for this test; SAR blending was used as it seemed more appropriate to keep the central template-based and peripheral saliency-based measures of conspicuity spatially distinct. A more extensive test of qualitative performance was conducted

[1]Average saccadic latency during scene viewing is 3-4 fixations per second (approximately 300ms [309]). Taking the upper bound of this average ensures that sufficient data is generated by our system to account for most individual variation.

(a) *Unexpected Visitor*



(b) Eye trace



(c) Predicted trace

Figure 5.5: Here the Unexpected Visitor (a.) is shown with a corresponding eye trace (b.) when the viewer was instructed to estimate the age of the people in the painting (a task for which faces are of particular importance). The predicted fixation sequence generated by STAR-FC using the set of face templates shown in Figure 5.4 as the conspicuity signal in the central field is shown in (c.) for comparison.

once the central field was generalized to be image independent, as detailed in the next section.

5.5.2 Extending the Central Field for Generality

Although the template search used in Section 5.5.1 serves as an initial demonstration for the STAR-FC architecture, it is a conceptually unsatisfying approach as well as impractical for any large-scale dataset, as it requires user specification of the expected fixation targets (at least with regards to the central field component). The release of the OpenSALICON [220] implementation of the SALICON [181] model provided an accessible code base for experimentation. Although they lack many of the original assumptions inherent to the pre-attentive model of saliency (such as shallow, fast computational architectures), deep learning saliency models provide a natural starting point for approximating the central field, as they can be viewed as analogous in featural complexity to a feed-forward sweep through the visual hierarchy.

As a qualitative test of the performance of STAR-FC using a generalized model of central saliency based on deep features, STAR-FC was re-run on Ilya Repin's *Unexpected Visitor* painting (Figure 5.3). Given the fact that SALICON relies on free-viewing training data, it made more sense for the predicted sequences produced by the generalized implementation of STAR-FC to be compared against the traces recorded by Yarbus that are similarly based on free-viewing periods of observation (Figure 5.6). As can be seen in these traces, there is a fair amount of individual variation, but there are nevertheless a number of commonalities in both fixation locations (including a focus on faces, though not to the same degree as seen in Figure 5.5) and the qualitative pattern of investigation (such as clusters of short saccades over regions of interest connected by larger exploratory saccades to a new region of interest).

In addition to the *Unexpected Visitor* the switch to free-viewing allowed comparison against two other paintings on which Yarbus had recorded eye traces under free-viewing conditions: Isaac Levitan's *The Birch Wood* (Figure 5.7a) and Ivan Shishkin's *Morning in a Pine Forest* (Figure 5.8a). Each of these additional paintings only had results reported by Yarbus for a single observer, and each for a different length of time [10]. Levitan's painting was viewed for ten minutes and Shishkin's for two; using the same assumption of four saccades per second as was used for the Repin painting, that means that 2400 fixations were calculated for the Levitan painting and 480 fixations for the Shishkin painting.

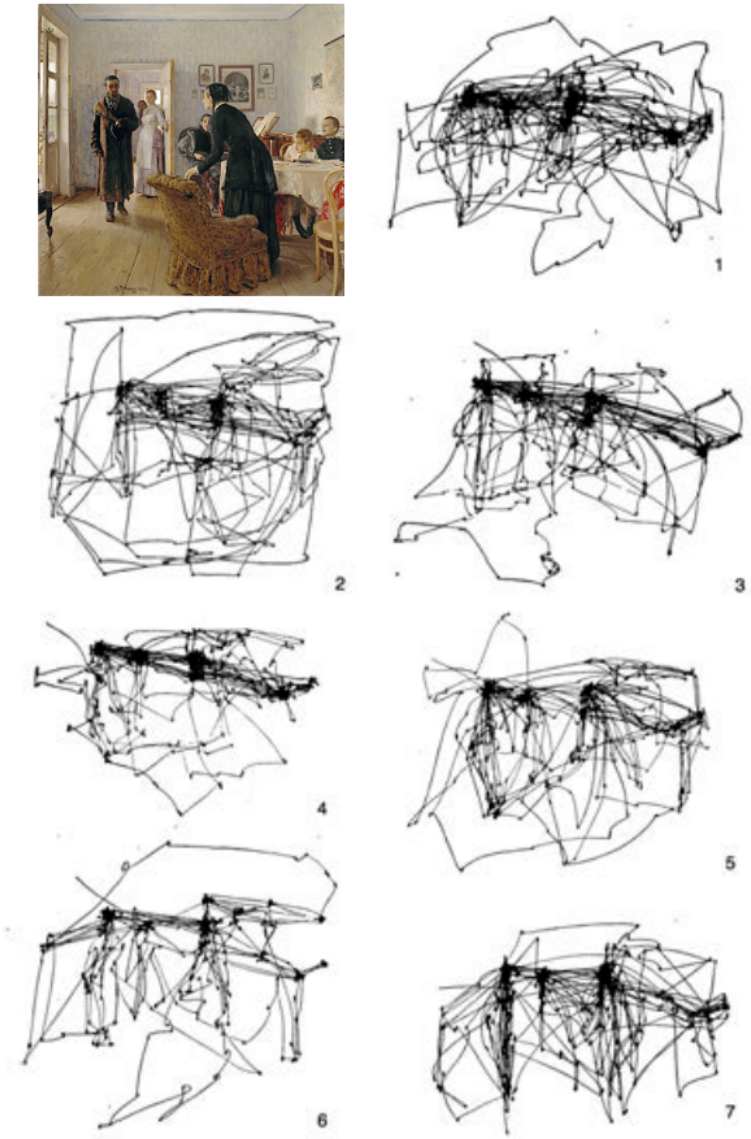
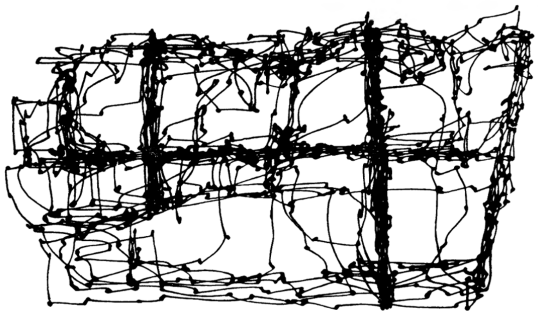


Figure 5.6: Fixation traces recorded by Yarbus [10] for seven individuals viewing Ilya Repin's Unexpected Visitor painting without task instructions (with the painting in the upper left for reference).

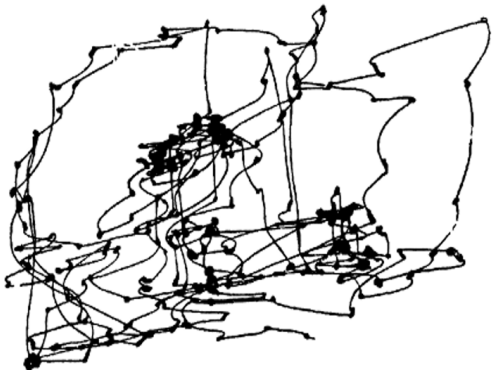
(a) *The Birch Wood*



(b) Eye trace

Figure 5.7: Isaac Levitan's *The Birch Wood* and a corresponding eye trace recorded by Yarbus [10] for an observer viewing the painting for ten minutes without task instructions.

(a) *Morning in a Pine Forest*



(b) Eye trace

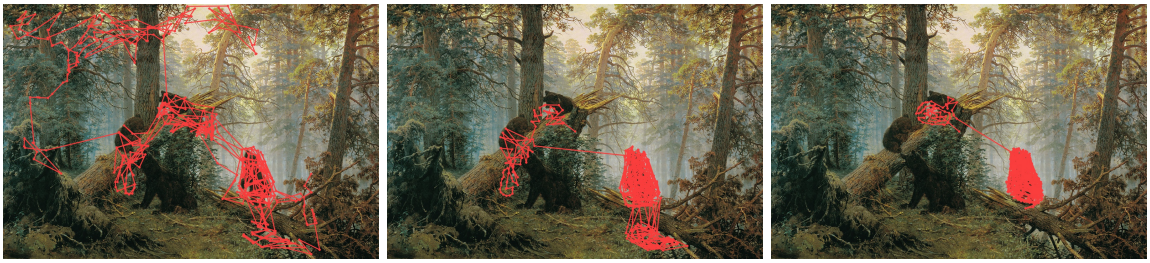
Figure 5.8: Ivan Shishkin's *Morning in a Pine Forest* and a corresponding eye trace recorded by Yarbus [10] for an observer viewing the painting for two minutes without task instructions.

One potential issue with directly using a model such as SALICON as the central field component of STAR-FC is the effect of the retinal transform on its performance. The SALICON model is trained on stimuli without blurring, and therefore may not handle the degredation of the retinal transform very well. Geirhos *et al.* [310] demonstrated that deep neural networks tend to degrade much more quickly in performance than human observers when faced with noise that has not been explicitly trained against. This may in part be rectified by some of the central-peripheral blending strategies described in Section 5.4.2 (this is explored in more detail in Section 5.7.1), but it may be worthwhile in the future to develop a central field network explicitly designed to function over an anisotropic visual field.

Yarbus did not report what viewing angle was covered by the stimuli he used in his experiments [10], meaning that this became an unspecified parameter in our tests of STAR-FC. Therefore, for each image, STAR-FC was run with a combination of three different parameters: peripheral-central blending strategy, peripheral gain factor, and viewing angle covered by the image. Peripheral-central blending was set to one of three choices, detailed in Section 5.4.2: *MCA*, *SAR*, or *WCA*. Peripheral gain, P_{gain} is a multiplicative factor applied to the peripheral map during the calculation of the conspicuity map; larger values of P_{gain} increase the likelihood of selecting a peripheral target and will therefore tend to lead to larger, more exploratory saccades. P_{gain} values used in these tests were sampled from 1.0 to 1.2 in increments of 0.05. Viewing angle covered by the image (based on its major axis) was set to 40°, 60°, 90°, or 120°. Thus, for each image, there are $[3 \times 5 \times 4] = 60$ total unique combinations of parameter settings that were tested.

The parameters do not always impact the pattern of predicted fixations independently, but we may nevertheless note some common trends for each, and qualitatively determine which sets produce the best match to the traces recorded for each painting. The parameter with the largest effect on performance appears to be the angle of view occupied by the image. For sufficiently large viewing angles, STAR-FC fails to explore the entire image and instead gets trapped in local interest points. This is most dramatically seen in Shishkin's painting (Figure 5.9), as the single cub standing against the bright backdrop of the white background haze in the lower right serves as a highly attractive draw that STAR-FC concentrates on more and more heavily until it completely fails to escape at 120° (Figure 5.9c). This is likely a combination of both the increased rate of peripheral image degredation for larger simulated fields of view, as well as the reduced size in pixels

of the IOR effect (IOR was set to a radius of 1.5°; this meant that for wider fields of view it would take more fixations over a specific number of pixels to fully inhibit them).



(a) 60° viewing angle **(b)** 90° viewing angle **(c)** 120° viewing angle

Figure 5.9: Fixation sequences generated by STAR-FC on Shishkin's painting for the MCA blending strategy with $P_{gain} = 1$ for three different sizes of view. As can be seen, wider angles of view tend to lead to less exploration of the whole image.

This pattern of more localized exploration as the extent of eccentricity covered by the painting increases is replicated on both the Repin (Figure 5.10) and Levitan (Figure 5.11) paintings. While it is probable that 40° or 60° of viewing angle more closely matches the display geometries used by Yarbus, human performance on wider fields of view would likely not be quite so strongly affected. As noted in Section 1.2.2.6, human fixations are also driven by endogenous strategies for information seeking, particularly for long fixation sequences. In its current form, the FHM component of STAR-FC only provides inhibition against returning to previously fixated locations, but it would provide the natural avenue for incorporating an information seeking strategy of spatial priority modulation.

(a) 60° viewing angle **(b)** 90° viewing angle **(c)** 120° viewing angle

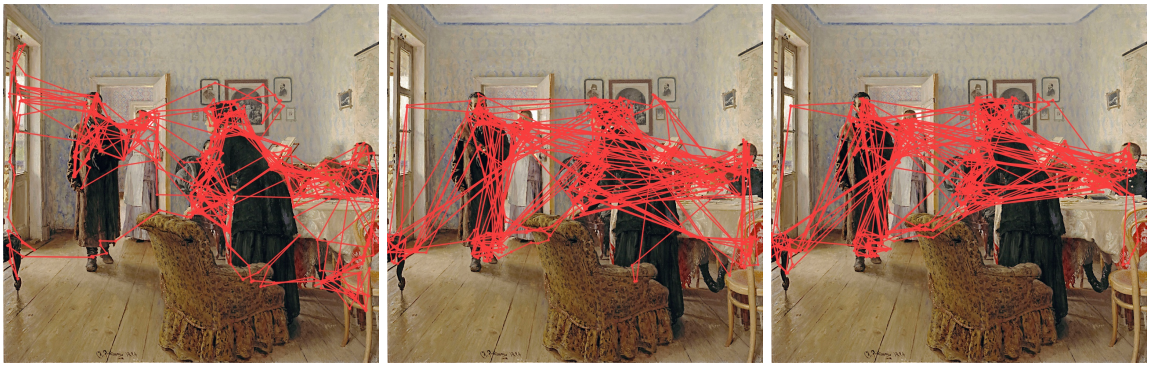
Figure 5.10: Fixation sequences generated by STAR-FC on Repin's painting for the SAR blending strategy with $P_{gain} = 1.05$ for three different sizes of view. As can be seen, wider angles of view tend to lead to less exploration of the whole image.

(a) 40° viewing angle **(b)** 90° viewing angle **(c)** 120° viewing angle

Figure 5.11: Fixation sequences generated by STAR-FC on Levitan's painting for the MCA blending strategy with $P_{gain} = 1.1$ for three different sizes of view. As can be seen, wider angles of view tend to lead to less exploration of the whole image.

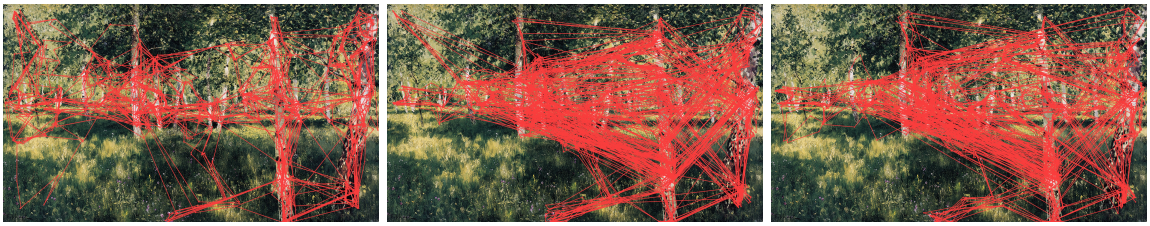
Although more subtle in effect than the angle of view, the central-peripheral blending strategy is the next most impactful parameter tested. The choice in strategy influences the spatial spread of conspicuity values, and will therefore be most apparent in the distribution of saccade amplitudes. Although SAR and WCA will not always lead to identical fixated regions (see, for example, Figure 5.14), they both appear to produce qualitatively similar amplitude distributions that skew saccadic amplitudes toward either very small or moderately sized jumps in fixation, leading to traces that appear somewhat choppy. MCA, in contrast, appears to produce a more evenly spread distribution of saccade amplitudes and therefore produces traces that are qualitatively more human-like than the other strategies. These differences are most readily apparent over a smaller field of view, and thus the examples shown in Figures 5.12, 5.13, and 5.14 are all chosen from 40° trials. Different representative P_{gain} values are chosen for each example to show that the observed qualitative patterns persist across P_{gain} values.

The P_{gain} value is the final parameter that was systematically varied for this test. Although there are clear subtle differences between for runs with different values of the P_{gain} parameter, it was often unpredictable what effect it might have. A good example of this is on the Shishkin painting at a viewing angle of 40° using the MCA blending strategy; as shown in Figure 5.15, many of the characteristics of the sequences themselves are similar, but there are nevertheless differences. In particular, the upper right corner is only fixated when $P_{gain} = 1$, the lower left stump is most covered when $P_{gain} = 1.05$, and the upper left is explored for $P_{gain} = 1.15$. These differences do not appear systematic nor predictable. The aim of the P_{gain} parameter is to provide a mechanism to balance larger, more exploratory jumps toward peripheral targets against shorter central-field



(a) MCA blending strategy **(b)** SAR blending strategy **(c)** WCA blending strategy

Figure 5.12: Fixation sequences generated by STAR-FC on Repin's painting with $P_{gain} = 1.05$ at a viewing angle of 40° for three different blending strategies. As can be seen, SAR and WCA are quite similar and qualitatively less human-like than MCA.



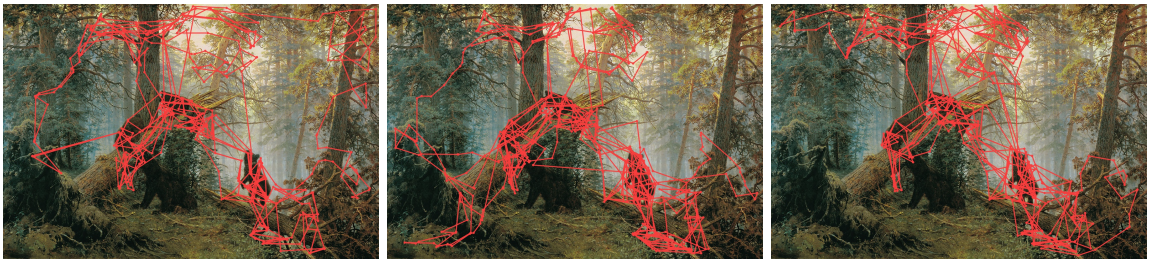
(a) MCA blending strategy **(b)** SAR blending strategy **(c)** WCA blending strategy

Figure 5.13: Fixation sequences generated by STAR-FC on Levitan's painting with $P_{gain} = 1.15$ at a viewing angle of 40° for three different blending strategies. As can be seen, SAR and WCA are quite similar and qualitatively less human-like than MCA.

(a) MCA blending strategy **(b)** SAR blending strategy **(c)** WCA blending strategy

Figure 5.14: Fixation sequences generated by STAR-FC on Shishkin's painting with $P_{gain} = 1.2$ at a viewing angle of 40° for three different blending strategies. As can be seen, SAR and WCA differ in some fixation locations, but in distribution of saccade amplitudes are qualitatively less human-like than MCA.

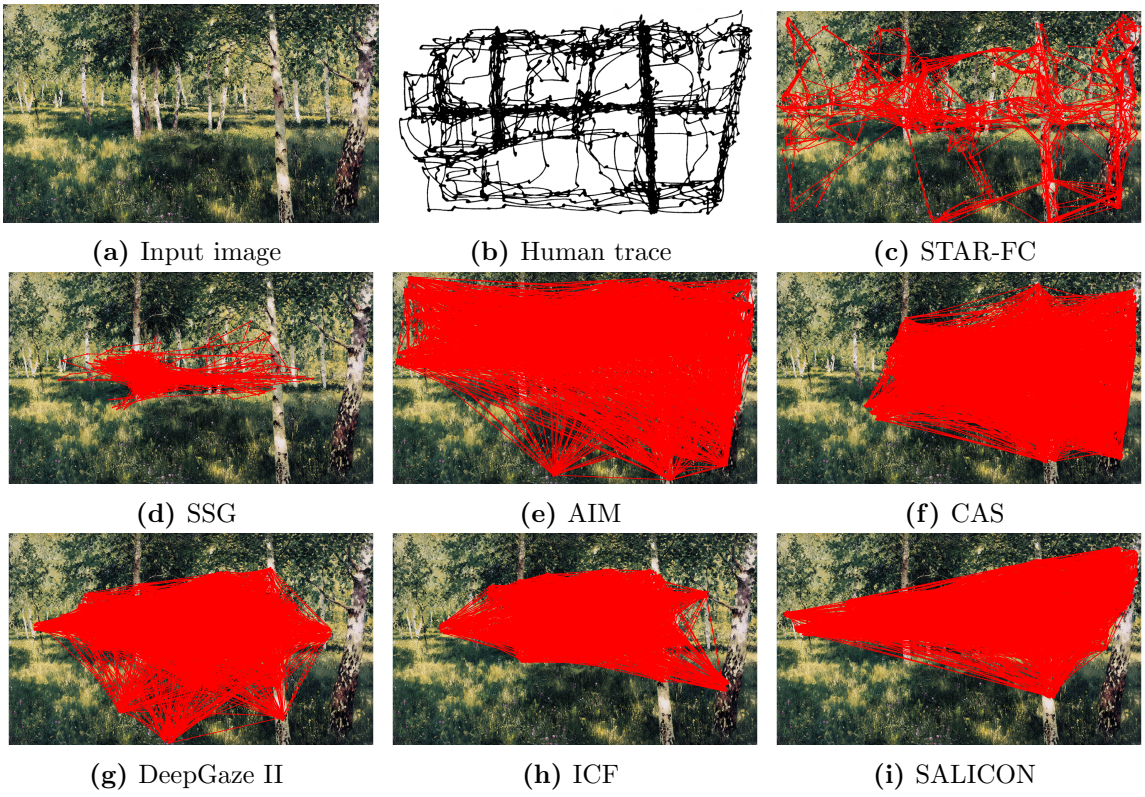
saccades, and this may be an important avenue for later development (in particular, a dynamic P_{gain} value might be an element in a visibility model style mechanism). However, when using a single static value, the range tested appeared to provide qualitatively similar results.



(a) $P_{gain} = 1.0$ **(b)** $P_{gain} = 1.05$ **(c)** $P_{gain} = 1.15$

Figure 5.15: Fixation sequences generated by STAR-FC on Shishkin's painting using the MCA blending strategy at a viewing angle of 40° for three different P_{gain} values. As can be seen, the gross characteristics are similar, but the specific sequences differ, particularly in terms of trajectories to the corners.

Overall, the set of parameters that qualitatively appear to provide the best performance across all three paintings tested by Yarbus is an MCA blending strategy at a 40° viewing angle and $P_{gain} = 1.05$. The sequences generated at these settings are shown in Figures 5.16-5.18, along with results generated by the SSG model [11] and by sampling from static saliency maps using an iterative WTA process with an IOR mechanism equivalent to the IOR of STAR-FC (1.5° radius with a linear decay over 100 fixations). Results are shown for AIM [2], CAS [143], DeepGaze II [182], ICF [184], and SALICON [181]. As can be seen, for all three images STAR-FC creates saccadic traces with a far more "human-like" appearance than the other methods. Unfortunately, direct quantitative comparison with Yarbus' output is not possible, as his methods were only capable of producing analog traces without accompanying coordinates. The next section, therefore, explores an alternative source of data in order to provide quantitative analysis of STAR-FC's performance.

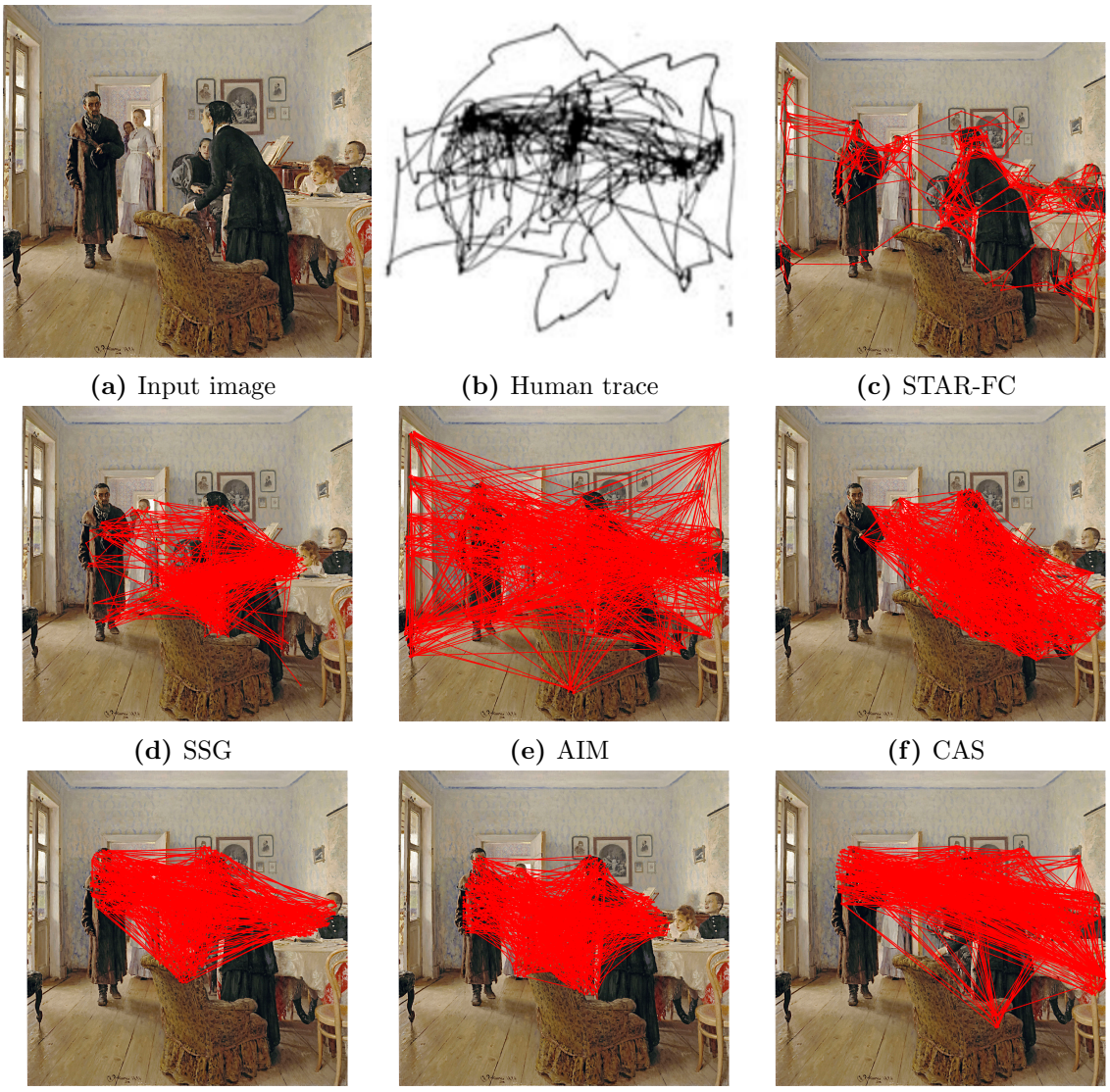


(a) Input image **(b)** Human trace **(c)** STAR-FC

(d) SSG **(e)** AIM **(f)** CAS

(g) DeepGaze II **(h)** ICF **(i)** SALICON

Figure 5.16: Fixation sequences generated by STAR-FC and a selection of competing methods on Levitan's painting. As can be seen, STAR-FC is much closer to the pattern of human fixations recorded by Yarbus than the sequences generated over static maps or generated by [11].



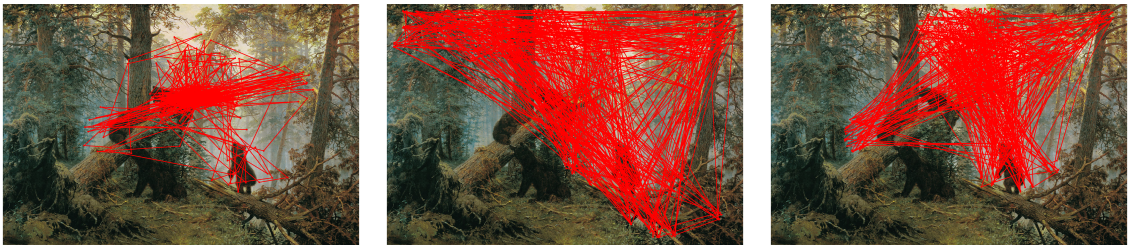
(a) Input image **(b)** Human trace **(c)** STAR-FC

(d) SSG **(e)** AIM **(f)** CAS

(g) DeepGaze II **(h)** ICF **(i)** SALICON

Figure 5.17: Fixation sequences generated by STAR-FC and a selection of competing methods on Repin's painting, with the first recorded human trace shown in 5.17b for ease of comparison (all human traces are displayed in Figure 5.6). As can be seen, STAR-FC is much closer to the pattern of human fixations recorded by Yarbus than the sequences generated over static maps or generated by [11].

(a) Input image **(b)** Human trace **(c)** STAR-FC



(d) SSG **(e)** AIM **(f)** CAS

(g) DeepGaze II **(h)** ICF **(i)** SALICON

Figure 5.18: Fixation sequences generated by STAR-FC and a selection of competing methods on Shishkin's painting. As can be seen, STAR-FC is much closer to the pattern of human fixations recorded by Yarbus than the sequences generated over static maps or generated by [11].

5.6 Quantitative Evaluation

5.6.1 Fixation Dataset

In order to provide a quantitative analysis of its performance, STAR-FC is evaluated over the CAT2000 dataset[2] [12]. This dataset was chosen due to several positive attributes: it contains twenty different image categories (thereby representing a wide spectrum of visual stimuli), as well as one of the widest fields of view that we are aware of for a free-viewing eye tracking dataset (approximately 45°). Larger fields of view better approximate natural scene exploration, and are also likely to be more greatly impacted by considerations of retinal anisotropy and motoric bias than a comparable dataset gathered over a narrower field of view.

5.6.2 Evaluation Metrics

One major challenge in this work was determining the best method for evaluation. The output of STAR-FC is not directly comparable to that of saliency algorithms designed to predict human fixations, as it outputs a sparse set of explicitly predicted locations rather than a smooth map that can be treated as a probability distribution for likely fixation points over an image [260]. However, as mentioned in Section 5.1.1, there are applications for which an explicit sequence of fixation points is preferable to a probabilistic heat-map that lacks temporal structure.

Given that the innovation of our work rests on providing an explicit, temporally ordered fixation sequence rather than on a novel representation of saliency, we focus on evaluation metrics that reflect the spatiotemporal structure of sequences. In order to compare against the static maps that are the standard output of saliency algorithms, we sampled fixation sequences from the maps by applying an iterative WTA procedure. IOR was applied to each selected location using the same parameters as those of our fixation control model. This technique is consistent with previous work that samples loci of attention from saliency maps [64], and is the same method used to generate long sequences for qualitative comparison in Figures 5.16-5.18.

Although saccade amplitude distributions provide a relatively coarse measure with which to compare fixation sequences (as there is no representation of positional differences over the visual field), they do provide a representation of the motoric bias in the prediction. An early criticism of

[2]Groomed in the same manner as in Section 4.3.

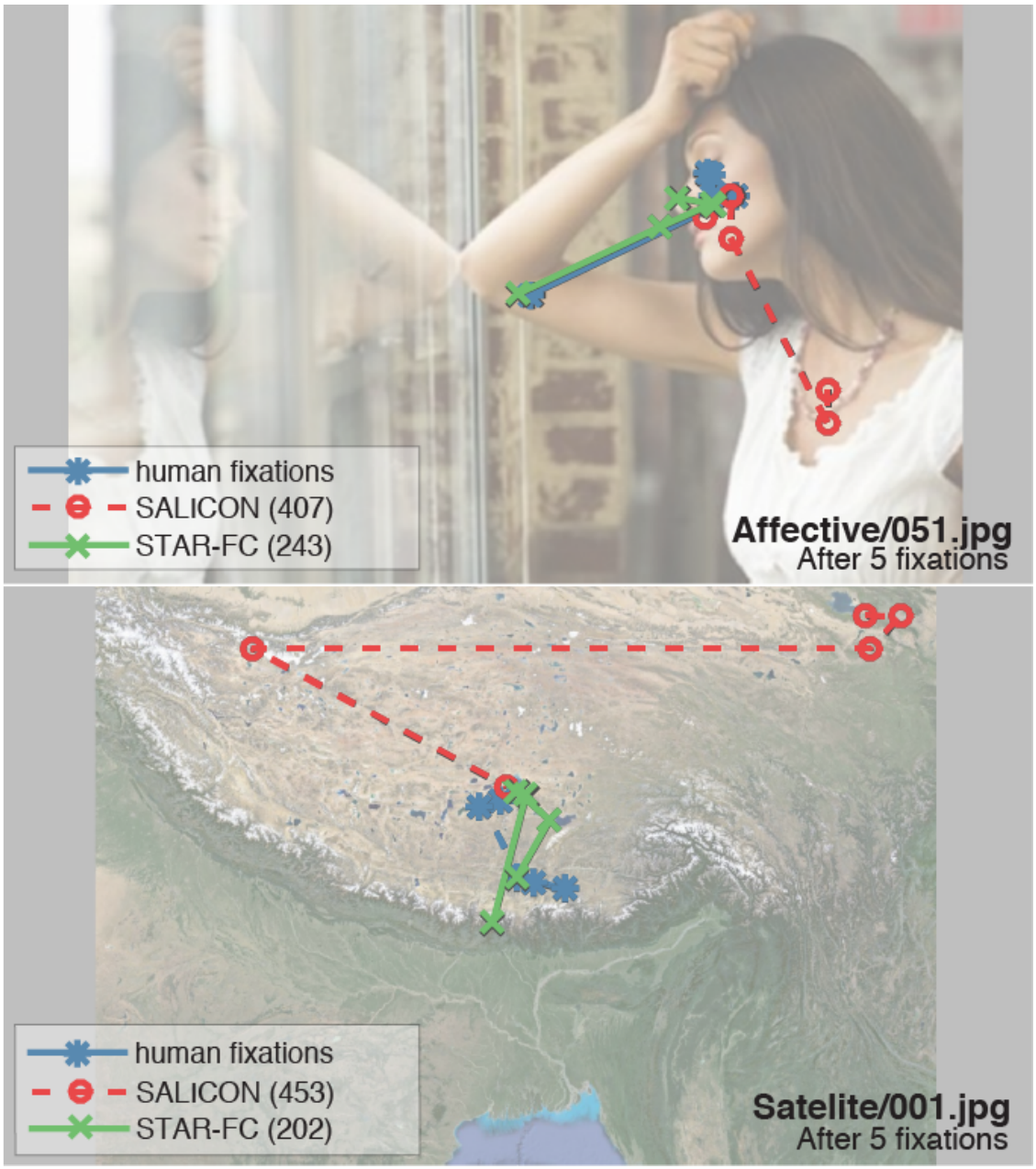


Figure 5.19: Two examples from the CAT2000 dataset with overlaid fixation sequences for the first five fixation points. The sequences predicted by the STAR-FC model are shown in green with X's marking the fixations, and SALICON predictions are shown in red with O's marking the fixation points. The human sequences that provided the closest match to each model are shown in blue. Euclidean distances between each model and the corresponding human sequence are noted in parentheses in the legend. Note that in both images, STAR-FC is much closer to human behaviour than SALICON.

saliency algorithms was that they fail to account for inherent motor biases in how humans move their eyes [170], and, as argued in Section 2.1.1, these inherent traits in saccadic vision may be an unavoidable contributing factor to centre bias. We therefore examine this aspect of model function in Section 5.7.1, demonstrating a much more human-like distribution of saccade amplitude with our model than is found from the predictions sampled from static saliency maps.

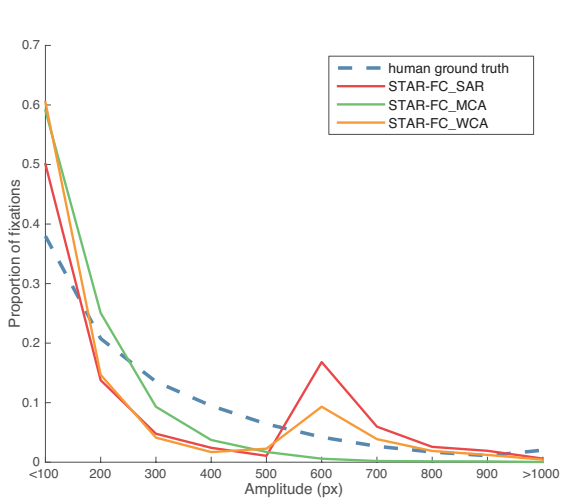
To more explicitly explore the prediction performance of our model, we utilize trajectory-based scoring methods. These metrics focus on measuring the deviation between two spatiotemporal sequences. Trajectory comparison is a common problem in a wide range of fields, and can often rely on a number of different constraints or assumptions. Three common classifications of trajectory metrics are *network-constrained*, *shape-based*, and *warping-based* [272]. Network-constrained methods rely on an underlying path structure (such as a road network), and were therefore not appropriate for our purposes. However, both shape-based (which measure the spatial structure of trajectories) and warping-based (which take into account the temporal structure as well as the spatial) can provide meaningful insight for saccadic sequences, and we therefore utilized the following set in order to provide a comprehensive sense of performance (trajectory-based score results are found in Section 5.7.2):

- *Euclidean Distance (ED)*: ED is one of the most common and basic warping-based trajectory metrics, and is calculated by matching two sequences in temporal order and computing the average pairwise distance between corresponding fixation points.

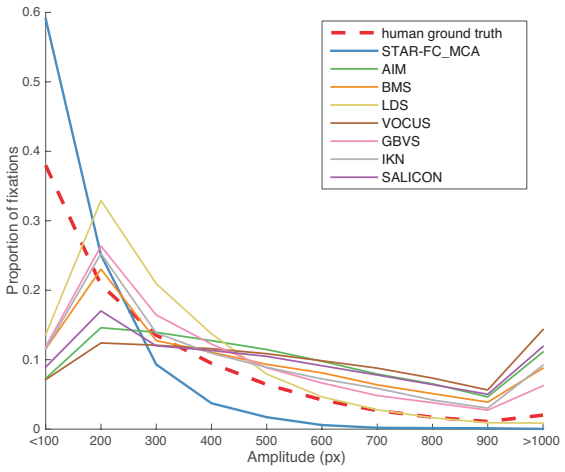
- *Fréchet Distance (FD)*: FD, also referred to as the 'dog-walking distance', represents the maximum distance at any given point in time over the length of two trajectories.

- *Hausdorff Distance (HD)*: HD is the maximum distance of a point in one sequence to the nearest point in a second sequence. Unlike ED and FD, HD is purely spatial and does not take sequence order into account.

Two example images from the CAT2000 dataset are shown in Figure 5.19, along with the first five fixations predicted by STAR-FC and SALICON. The closest matching human sequence is provided in blue. Along the visual comparison of their output, the image legends also provide the Euclidean Distance score between each predicted sequence and the shown human sequence.

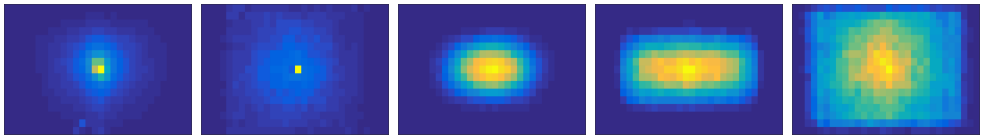


(a) STAR-FC variants compared to humans



(b) Traditional saliency algorithms, STAR-FC, and human distributions

Figure 5.20: Plots of the saccadic amplitude distributions over the CAT2000 dataset. Saccade lengths were assigned to bins of pixel ranges and the proportion of saccades falling in each bin are shown in the figures: (a) shows the effect of the different STAR-FC configurations on the resultant saccadic amplitude distribution (contrasted with the human distribution shown with a dashed line); (b) shows the distributions of traditional saliency algorithms contrasted with the MCA variant of STAR-FC and the human distribution.



(a) Human (MSE Score)

(b) STAR-FC, (0.002)

(c) LDS, (0.027)

(d) GBVS, (0.072)

(e) SALICON, (0.122)

Figure 5.21: 2D histograms of fixation locations over the CAT2000 dataset. Mean-squared-error (MSE) scores between model and human distributions are shown in parentheses under each model name; as can be seen, STAR-FC is an order of magnitude closer to the human distribution than the closest competing saliency model.

5.7 Results

We compare the performance of STAR-FC with a range of established saliency models: AIM [154], BMS [307], GBVS [78], LDS [165], SALICON [7, 220], SSR [161], and VOCUS2 [150].

5.7.1 Spatial Distributions

Saccadic amplitude distributions are shown in Figure 5.20. As can be seen in Figure 5.20a, the original central-peripheral integration strategy of Separate Activation Regions (SAR) used Section

5.5.1 has a tendency to create a bimodal distribution not seen in the human data. This is likely due to the fact that the retinal anisotropy creates a biased gradient to the output of both the central and peripheral fields, meaning that near the border of the two the central field is weakest and the peripheral field is strongest. In order to facilitate a smoother transition of activation across the visual field, we tested two other integration strategies (described in Section 5.4.2): Maximum Central Activation (MCA) and Weighted Central Activation (WCA).

Our motivation to allow for the low-level feature representation of the peripheral map to affect the central region but not the other way around is based on the fact that there do appear to be fundamental perceptual limitations in object perception and feature binding within peripheral vision [274], whereas low-level features do seem to have a persistent role in attentional guidance [184].

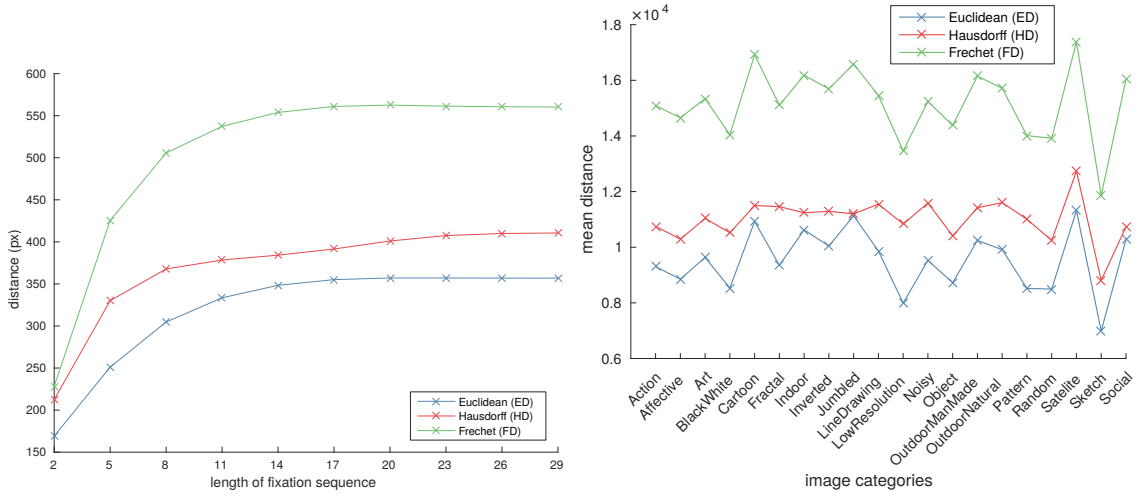
Despite blending peripheral and central activations in a smoothly merging fashion, the WCA strategy leads to an activation pattern remarkably similar to the original SAR strategy. This is likely due to the fact that a weighted blending penalizes the chances of both algorithms within the mid-central region to attract attention unless they both happen to achieve a high score, essentially requiring a target to attract both high and low level attention simultaneously.

The closest distribution pattern to that of humans was achieved by the MCA integration strategy, and it is therefore the variant reported in Figure 5.23 and Table 5.1. Although it does match the human distribution more closely than WCA and SAR variants, MCA appears to over-emphasize short saccades, having a much shallower tail than seen in the distribution of human observers. As previously mentioned, one likely contribution to this over-emphasis is the difficulty of many algorithms that have not been explicitly designed or trained to deal with signal degradation to function effectively across the retinal transform.

In contrast to the STAR-FC amplitude distributions, virtually all static saliency maps are skewed in the opposite direction with distributions that are much flatter than those seen with human data. Many algorithms do retain a small preference for shorter saccades, but this could also be an outcome of compositional bias in the underlying images. 2D histograms of fixation location produced with 64×64 sized blocks across the full CAT2000 dataset are shown for humans along with the MCA variant of STAR-FC and several representative saliency algorithms in Figure 5.21. As can be seen, there does appear to be a consistent spatial bias toward the centre of the image

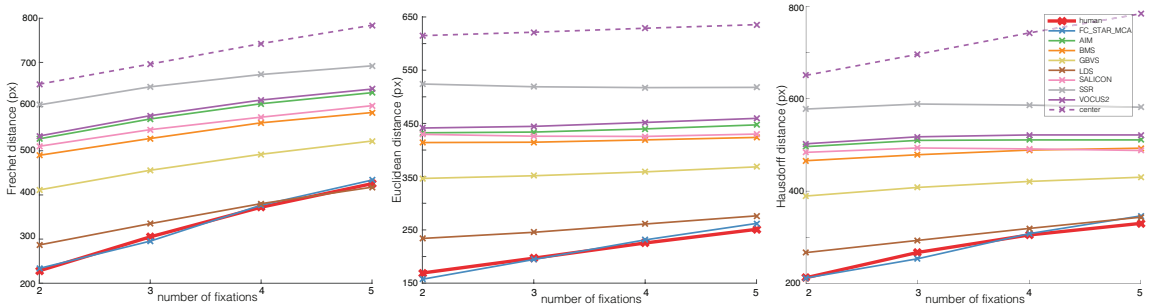
that, at least in part, likely represents the underlying composition of the dataset images. Likewise, the saliency algorithms with the closest spatial distribution to the human distribution do tend to have a greater propensity for shorter saccades (Figure 5.20).

5.7.2 Trajectory Scores



(a) Average scores with sequence length **(b)** Average total sequence score by category

Figure 5.22: Average scores computed for all metrics over pair-wise matches of human sequences: (a) shows that as sequence length increases observer agreement tends to diverge, leading to a saturation in score values for each metric; (b) shows average sequence score per category, showing agreement with [12] about which categories tend to have greatest inter-observer consistency.



(a) Mean FD **(b)** Mean ED **(c)** Mean HD

Figure 5.23: A comparison of fixation prediction scores for static saliency maps and STAR-FC. A sequence formed by always picking the centre pixel is shown in a dashed line to provide a performance baseline.

Figure 5.22 shows the results of computing pair-wise scores across all combinations of human

Model	AUC ED	AUC HD	AUC FD	MSE
Human	632	844	1004	0
STAR-FC	**630**	**841**	**1006**	**0.002**
LDS	762	918	1067	0.027
GBVS	1068	1239	1415	0.072
BMS	1253	1447	1629	0.102
SALICON	1281	1471	1680	0.122
AIM	1313	1525	1758	0.161
VOCUS2	1347	1551	1781	0.183
SSR	1557	1755	1966	0.183
center	1875	2156	2156	0.008

Table 5.1: Algorithm performance. Area-under-the-curve (AUC) scores are reported over the first five fixations for each plot in Figure 5.23. Note that our model (in bold) matches the inter-subject error of human observers. The last column shows the mean-square-error for the spatial histogram of predicted fixations versus the distribution of human fixations over the entire dataset.

sequences for each image from the CAT2000 dataset. Figure 5.22a shows that the different trajectory metrics all tend to drift toward a saturated value; ED, FD, and HD all get larger as sequences diverge through time. Additionally, it has been shown that early fixations tend to be most similar across subjects [95], and inter-observer consistency degrade largely after the first five fixations. We therefore restrict our analysis to only this interval. Analysis of the full sequences may be found in Appendix B.

Figure 5.22b shows the category-wise average total sequence scores per category. Here we can see that the trajectory metrics largely agree with the analysis in [12] on which categories have the greatest inter-observer consistency (Sketch, Low Resolution, and Black and White), and which categories tend to have poor inter-observer consistency (Satellite, Jumbled, and Cartoon).

We compare STAR-FC against a wide selection of saliency algorithms in Figure 5.23, showing that STAR-FC consistently achieves trajectory scores more in line with human sequences over the critical range of the earliest fixations, followed by LDS and GBVS (see Table 5.1 for numerical scores). In fact, STAR-FC is the only model that is able to achieve near-parity with the natural heterogeneity found within human observers.

As is made clear in Figure 5.21, human fixations over CAT2000 are strongly biased toward the centre, a distribution that is well-matched by STAR-FC. The best performing saliency algorithms (LDS [165] and GBVS [78]) likewise have correspondingly stronger biases toward predicting fixations

near the image centre. We therefore also tested the "center" model, which is simply a sequence that always selects the central pixel for every fixation. This selection will minimize the upper error bound for all trajectory metrics, and can be qualitatively thought of as a similar performance baseline to a centered Gaussian for more traditional saliency metrics [4]. Nevertheless, as Figure 5.23 shows, the center model consistently achieves the worst score in all metrics, confirming that while a centrally focused distribution of fixation locations is appropriate for the CAT2000 dataset, it is not a sufficient characteristic to score well.

5.8 Conclusion

STAR-FC provides a powerful tool for predicting explicit fixation sequences. demonstrating fidelity to human fixation patterns equivalent to that of using one person's fixation sequence to predict another. This performance is significantly better than what can be achieved by sequence sampling from static saliency maps (see Table 5.1). Our model will allow improved performance in saliency applications relying on explicit fixation prediction, including for commercial [286] and science communication [288] purposes. In addition to its performance, our model is also constructed to provide a descriptive model of fixation control, allowing further research into the interaction of the different cognitive control architectures that link gaze to higher order visual cognition [98].

While it is clear that retinal anisotropy has a significant effect on human visual performance, very few computational algorithms are developed with the aim of dealing with anisotropic acuity. This creates a significant challenge to accurately detect and ascribe conspicuity values across the visual field, and our model's incorporation of retinal anisotropy represents an interesting platform for exploring this area of research.

Additionally, free-viewing over static images represents only a very narrow range of task for which fixation prediction provides valuable information. Fixation prediction over video and under task demands are highly challenging domains for which explicit fixation control may prove extremely valuable.

Chapter 6

Conclusions and Future Directions

6.1 Summary of Contributions

This dissertation reviews the development and history of computational saliency modelling and presents three distinct research threads: a critical analysis of performance metrics (Chapter 2), a unified library of models with a common implementation format (Chapter 3), and a novel model of human fixation control that outperforms static saliency maps at saccade sequence prediction (Chapter 5). Chapter 2 demonstrates that human fixation patterns have a natural spatial component to them and argues that this may not be separable from the visual saliency component. Saliency metrics that attempt to correct for this spatial bias themselves introduce assumptions and biases, and may obscure performance characteristics of the models being tested. This risks encouraging algorithm development that designs to the test, and may not reflect the truly desired performance characteristics for the task of fixation prediction and understanding human eye movement decisions. A novel metric, spROC, is introduced that provides a clear assessment of algorithm spatial bias.

Chapter 3 presents an implementation library that wraps a number of available saliency models in a common implementation format, and provides a template for further extension with future models. This was showcased in Chapter 4 with a test of performance for a number of saliency algorithms with several novel experiments exploring the performance of saliency models on psychophysical stimuli and visual search arrays that lead to asymmetric performance in humans, as well as the spatiotemporal patterns of saliency model predictions over natural images.

Chapter 5 uses saliency as part of a larger gaze control network, extending its application beyond a static saliency map to explicitly generate a temporally ordered sequence of fixations. This gaze control network, dubbed STAR-FC, is shown to produce fixation sequences that are qualitatively much more similar to human eye movement traces, and quantitatively outperforms competing methods at predicting fixation trajectories on the CAT2000 dataset.

Altogether, all three threads engage with the prevailing approach to saliency research (namely, the development of saliency models for producing saliency maps) and presents a critical evaluation of the limits to this approach. Rather than treating saliency modelling as a numerical exercise of performance optimization, this work has sought to identify and forge ties between saliency model performance and broader aspects of attention, cognition, and computer vision applications.

6.2 Significance

Chapter 2 makes clear a number of issues with standard performance metrics, particularly the widely used *shuffled* ROC method, and makes clear that centre bias in human fixation data is not a factor that can simply be worked around, but rather is an intrinsic issue with freeviewing over natural images. This motivates a deeper exploration of the aspects involved in fixation targetting and saccadic control, which is developed in Chapter 5. Additionally, the degree of spatial bias in a number of commonly used saliency algorithms is clearly elucidated, providing guidance for the performance characteristics of these algorithms that may be useful when selecting a saliency algorithm for a novel task.

Chapter 3 further supports the application of saliency beyond standard fixation prediction benchmarks by facilitating rapid experimentation and testing of saliency models. Additionally, this allows more easily for the reproduction of benchmarking efforts, which is an important consideration for scientific research, and a notable challenge in fields reliant on software models [311]. Likewise, for research efforts that are not focused on the development of saliency models, but which nevertheless seek to use them, to come to some conclusion about the role or impact of saliency, it is common for only a single model to be tested (*e.g.*, see [52, 235]) or a small number of models (*e.g.*, see [312]). SMILER will allow for a broader set of algorithms to be tested, enhancing the robustness of claims made regarding the role of saliency in general as opposed to the efficacy of specific models. Several experiments showcasing this application are presented in Chapter 4, which reveal a deficit in performance for deep learning models over psychophysical stimuli and the prediction of human visual search behaviour, suggesting that training over natural images alone is insufficient to learn a complete account of human saliency representation.

Chapter 5 presents the STAR-FC model, which provides a significant improvement over previous methods for predicting explicit saccadic sequences and demonstrates the importance of accounting for spatial anisotropy in terms of both retinal sensitivity and perceptual processing. STAR-FC is designed with a strong consideration for biological fidelity, and provides a model that not only has strong predictive performance but also clear explanatory power for experimentation.

Both SMILER and STAR-FC represent complementary approaches to extending saliency research to link its findings to broader cognitive models and psychological theories. Despite the heavy

focus on fixation prediction in standard saliency benchmarks, fixations may be directed by a number of considerations outside of the visual salience of scene elements and purely data-driven approaches may overfit to the task of fixation prediction rather than the modelling of saliency. SMILER facilitates the exploration of other properties and behavioural predictions of current saliency models (such as the correlation of saliency values with human response times), thereby ensuring a more comprehensive evaluation of saliency model output with a focus a fuller understanding visual saliency. STAR-FC provides context for saliency map output, linking saliency maps to a larger cognitive control model with semantic meaning. This opens the door for the explicit inclusion of other considerations for fixation allocation without conflating these gaze control components.

6.3 Future Directions

This dissertation raises a number of possible avenues for future research directions. Though not an exhaustive list, a number of promising directions are reviewed here.

6.3.1 Saliency Experimentation Empowered by SMILER

Sections 4.1-4.3 provide case studies of the use of SMILER to rapidly perform large-scale testing and experimentation with saliency algorithms, but these can easily be extended further. As was seen in Section 4.1, some saliency models struggled to highlight oriented bar singletons even when the orientation difference was well within the range of pop-out. Therefore, it would be worthwhile to extend this test to a comprehensive set of basic features, including, for example, colour, vernier offset, and size (see [15] for a more complete discussion of basic features). All these features are clearly salient to human observers, and identifying those that are well represented by saliency algorithms and those that are not will not only provide a principled scoring mechanism for evaluating algorithm performance, but may also identify future directions of development for saliency models.

There are a number of additional nuances to the human judgement of salience that are worth exploring. For example, Nothdurft [53] showed that when faced with targets defined by salience along different feature dimensions, human judgements of the *most* salient target do not necessarily indicate an equal contribution of the different feature dimensions. A principled testing of how well different saliency models match human judgements of relative salience between multiple targets

will again provide another avenue for evaluation, while also directly pointing at a possible method for model improvement through a principled re-weighting of feature relevance. Should it be found that learned models do not well match the relative judgements of human subjects, it may help identify either deficiencies in the feature content of their training data or could point to limitations of the foundations on which the models are based. Likewise, while Nothdurft has tested relative contributions to the judgement of saliency for a subset of basic features, there are many additional basic features identified by Wolfe [15] that have not been compared for relative performance in humans. It may be interesting to look for consistent rankings of importance from saliency models as a prediction of human behaviour that may then be experimentally tested.

The testing of response asymmetry on complex objects found in Section 4.2.3.2 was carried out on a set of exemplars from the person class based on the strong representation of this class both in the training data for the learned features in deep learning networks as well as in the pool of targets for human fixation data. However, it would be interesting to extend this analysis to other categories, perhaps ones that are of semantic interest but are nevertheless not nearly so well-represented in fixation data as human targets. Discovering whether the opposite direction of asymmetry found in learned models is consistent across training categories or rather represents an overfitting in the case of the person category would be a useful discovery in the evaluation of this modern approach to saliency modelling.

Section 4.3 extended the evaluation based on spatial binning presented in Chapter 2 to the temporal domain, discovering some surprising spatiotemporal patterns of fixation correlation with saliency. Repeating this analysis across additional fixation datasets in natural images would be useful to help ensure that they are not artifacts of the particular data gathering techniques used in the generation of the CAT2000 dataset. Likewise, extending this analysis to datasets with different image types or tasks may yield a highly fruitful set of investigations. For example, in the visual search task conducted by Neider and Zelinsky [109], it was found that human observers would base their early fixations on likely target locations determined by scene context, but once those regions were exhausted would rapidly search a number of unlikely target locations for added assurance that the target truly was absent. An open question remains as to how the priority of these unlikely locations is decided; is it largely informed by saliency, or does it still follow a strategic ordering more consistent with a visibility model? By splitting fixations into temporal bins and selectively

analyzing these late fixations for correlation with saliency, it may be possible to gain insight into this matter.

Additionally, the spatiotemporal analysis conducted in Section 4.3 was on data gathered over static stimuli. Repeating this analysis over dynamic stimuli would be highly informative in the determination of how quickly narrative effects begin to take hold, or perhaps to discover some as yet unexpected correlations between behavioural patterns and stimulus saliency.

6.3.2 STAR-FC: Extensions and Experiments

The STAR-FC model presented in Chapter 5 is explicitly designed to be extensible within the cognitive programs framework proposed by Tsotsos and Kruijne [98]. Foremost among these extensions would be to extend the representation of spatial attentional pull (which currently only incorporates the inhibitory influence of IOR) with an excitatory mechanism consistent with the information gathering strategies proposed by visibility models. Including such a mechanism will likely help modulate the sometimes drastic changes in fixation patterns produced by STAR-FC when run at different simulated eccentricities (*e.g.* Figure 5.11), and may yield unexpected and informative interactions between information gathering strategies and the pull of stimulus-driven conspicuity.

Additionally, the current formulation of STAR-FC makes use of well-established saliency models to represent the contributions of both the central and peripheral fields. The use of these models helps anchor STAR-FC's performance within the larger literature on saliency modelling, but, as noted in Chapter 5, none of these models were originally designed for use in the presence of retinal anisotropy. Therefore, it would be beneficial to develop a novel set of saliency models that are explicitly intended to function within STAR-FC's architecture over an anisotropic field of visual acuity. Likewise, with learning-driven models of sequence prediction becoming more common and powerful (*e.g.* [301], [300]), it would be instructive to benchmark the performance of STAR-FC against these performance-driven models, particularly over a range of datasets and data gathering conditions.

While the tests of STAR-FC presented in Chapter 5 are over static stimuli in order to aid comparison within the predominant paradigm of fixation prediction, STAR-FC incorporates a built-in temporal component that would make it a prime candidate for development and testing over

dynamic stimuli. In order to do so the conspicuity representations would need to be extended to incorporate motion features. Dynamic stimuli would also encourage the extension of STAR-FC's eye movement behaviour beyond just saccadic jumps to also incorporate smooth pursuit. As mentioned in Section 5.1, there are a number of different models of eye movements, but it is rare for one to integrate control for more than one type into a single cohesive model. Adding this functionality to STAR-FC, therefore, would be a major step toward a more complete model of human eye movement control.

Finally, the dynamic nature of STAR-FC makes it a strong starting point for the control of fixations in an active vision system (such as the TRISH head [313]). Open-ended perceptual control for an embodied system would be very useful for the development of independent robotic agents. Placing STAR-FC on an instantiated optomechanical system would allow for a number of the simulated aspects of the model (such as retinal anisotropy, see [290] for a possible example of how) to be instead built into the system design. The integration of STAR-FC in this manner will thereby further our understanding of the principles of active vision.

6.3.3 Saliency and General Visual Attention

As mentioned in Chapter 1, saliency represents a subfield of visual attention, and an important outstanding scientific goal remains a full accounting of general visual attention. One of the primary motivations of this work has been to better situate saliency research within this broader picture of visual attention, but there is far more to this task which lies beyond the scope of this dissertation. Nevertheless, the tools and findings of this work will facilitate further advances in this field, including, for example, supporting research into the role and interaction of fixation control with visual working memory [314].

Bibliography

[1] T. Judd, F. Durand, and A. Torralba, "A benchmark of computational models of saliency to predict human fixations," Tech. Rep. MIT-CSAIL-TR-2012-001, Massachusetts Institute of Technology, 2012.

[2] N. Bruce and J. Tsotsos, "Saliency based on information maximization," in *Advances in Neural Information Processing Systems (NIPS)* (Y. Weiss, B. Schölkopf, and J. C. Platt, eds.), pp. 155–162, 2006.

[3] E. Vig, M. Dorr, and D. Cox, "Large-scale optimization of hierarchical features for saliency prediction in natural images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[4] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *IEEE International Conference on Computer Vision (ICCV)*, September 2009.

[5] I. van der Linde, U. Rajashekar, A. C. Bovik, and L. K. Cormack, "DOVES: A database of visual eye movements.," *Spatial Vision*, vol. 22, no. 2, pp. 161–177, 2009.

[6] S. P. Arun, "Turning visual search time on its head," *Vision Research*, vol. 74, pp. 86–92, 2012. Special Issue on Visual Attention.

[7] X. Huang, C. Shen, X. Boix, and Q. Zhao, "SALICON: Reducing the semantic gap in saliency prediction by adapting deep neural networks," in *IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[8] N. D. B. Bruce, C. Catton, and S. Janjic, "A deeper look at saliency: Feature contrast, semantics, and beyond," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[9] S. Rahman and N. Bruce, "Saliency, scale and information: Towards a unifying theory," in *Advances in Neural Information Processing Systems (NIPS)* (C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, eds.), pp. 2188–2196, 2015.

[10] A. L. Yarbus, *Eye Movements and Vision*. New York, NY, USA: Plenum Press, 1967.

[11] O. Le Meur and Z. Liu, "Saccadic model of eye movements for free-viewing condition," *Vision Research*, vol. 116, pp. 152–164, 2015.

[12] A. Borji and L. Itti, "CAT2000: A large scale fixation dataset for boosting saliency research," *CVPR 2015 workshop on "Future of Datasets"*, 2015.

[13] D. J. Simons and D. T. Levin, "Change blindness," *Trends in Cognitive Sciences*, vol. 1, no. 7, pp. 261–267, 1997.

[14] Y. H. Zhou, J. B. Gao, K. D. White, I. Merk, and K. Yao, "Perceptual dominance time distributions in multistable visual perception.," *Biological cybernetics*, vol. 90, no. 4, pp. 256–63, 2004.

[15] J. M. Wolfe, "Visual search," in *Attention* (H. Pashler, ed.), East Sussex, UK: Psychology Press, 1998.

[16] M. I. Posner and Y. Cohen, "Components of visual orienting," in *Attention and Performance*, vol. X, pp. 531–556, London, UK: Erlbaum, 1984.

[17] G. Rizzolatti, L. Riggio, and B. M. Sheliga, "Space and selective attention," in *Attention and Performance*, vol. XV, pp. 231–265, Cambridge, MA, USA: MIT Press, 1994.

[18] J. K. Tsotsos, S. M. Culhane, W. Y. K. Wai, Y. Lai, N. Davis, and F. Nuflo, "Modeling visual attention via selective tuning," *Artificial Intelligence*, vol. 78, no. 1-2, pp. 507 – 545, 1995. Special Volume on Computer Vision.

[19] J. K. Tsotsos, *A Computational Perspective on Visual Attention*. Cambridge, MA, USA: MIT Press, 2011.

[20] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, no. 4, pp. 219–227, 1985.

[21] A. Treisman and G. Gelade, "A feature integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, 1980.

[22] J. J. Clark and N. J. Ferrier, "Modal control of an attentive vision system," in *IEEE International Conference on Computer Vision (ICCV)* (R. Bajcsy and S. Ullman, eds.), pp. 514–523, September 1988.

[23] P. A. Sandon, "Simulating visual attention," *Journal of Cognitive Neuroscience*, vol. 2, no. 3, pp. 213–231, 1990. PMID: 23972045.

[24] S. M. Culhane and J. K. Tsotsos, "An attentional prototype for early vision," in *European Conference on Computer Vision (ECCV)*, pp. 551–560, May 1992.

[25] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[26] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 37–44, June 2004.

[27] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 185–198, 2010.

[28] C.-K. Chang, C. Siagian, and L. Itti, "Mobile robot vision navigation and localization using gist and saliency," in *IEEE Conference on Intelligent Robots and Systems (IROS)*, October 2010.

[29] R. Roberts, D.-N. Ta, J. Straub, K. Ok, and F. Dellaert, "Saliency detection and model-based tracking: A two part vision system for small robot navigation in forested environments," in *The Proceedings of SPIE 8387*, April 2012.

[30] A. Rasouli and J. K. Tsotsos, "Visual saliency improves autonomous visual search," in *Canadian Conference on Computer and Robot Vision (CRV)*, May 2014.

[31] P. E. Scalf and D. M. Beck, "Competition in visual cortex impedes attention to multiple items," *Journal of Neuroscience*, vol. 30, no. 1, pp. 161–169, 2010.

[32] I. D. Gilchrist, "Saccades," in *The Oxford Handbook of Eye Movements* (S. P. Liversedge, I. D. Gilchrist, and S. Everling, eds.), ch. 5, pp. 85–94, Oxford, UK: Oxford University Press, 2011.

[33] C. A. Curcio, K. R. Sloan, R. E. Kalina, and A. E. Hendrickson, "Human photoreceptor topography," *Journal of Comparative Neurology*, vol. 292, no. 4, pp. 497–523.

[34] C. A. Curcio and K. A. Allen, "Topography of ganglion cells in human retina," *Journal of Comparative Neurology*, vol. 300, no. 1, pp. 5–25, 1990.

[35] H. von Helmholtz and translated by J.P.C. Southall, *Helmholtz's treatise on physiological optics*. Dover, translated from the 3rd German ed., 1962.

[36] C.-H. Juan, S. M. Shorter-Jacobi, and J. D. Schall, "Dissociation of spatial attention and saccade preparation," *Proceedings of the National Academy of Sciences*, vol. 101, no. 43, pp. 15541–15544, 2004.

[37] J. E. Hoffman and B. Subramaniam, "The role of visual attention in saccadic eye movements," *Perception & Psychophysics*, vol. 57, no. 6, pp. 787–795, 1995.

[38] A. R. Hunt and A. Kingstone, "Covert and overt voluntary attention: linked or independent?," *Cognitive Brain Research*, vol. 18, no. 1, pp. 102–105, 2003.

[39] R. M. Pritchard, W. Heron, and D. O. Hebb, "Visual perception approached by the method of stabilized images," *Canadian Journal of Psychology*, vol. 14, no. 2, pp. 67–77, 1960.

[40] E. Kowler, "Eye movements: The past 25 years," *Vision Research*, vol. 51, no. 13, pp. 1457–1483, 2011.

[41] M. Rucci and M. Poletti, "Control and functions of fixational eye movements," *Annual Review of Vision Science*, vol. 1, no. 1, pp. 499–518, 2015. PMID: 27795997.

[42] Z. M. Hafed, C.-Y. Chen, and X. Tian, "Vision, perception, and attention through the lens of microsaccades: mechanisms and implications," *Frontiers in Systems Neuroscience*, vol. 9, no. 167, 2015.

[43] A. Borji, "Boosting bottom-up and top-down visual features for saliency estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 438–445, June 2012.

[44] H. Tian, Y. Fang, Y. Zhao, W. Lin, R. Ni, and Z. Zhu, "Salient region detection by fusing bottom-up and top-down features extracted from a single image," *IEEE Transactions on Image Processing*, vol. 23, pp. 4389–4398, Oct 2014.

[45] G. Zhu, Q. Wang, and Y. Yuan, "Tag-saliency: Combining bottom-up and top-down information for saliency detection," *Computer Vision and Image Understanding*, vol. 118, pp. 40–49, 2014.

[46] D. J. Felleman and D. C. Van Essen, "Distributed hierarchical processing in the primate cerebral cortex," *Cerebral Cortex*, vol. 1, no. 1, pp. 1–47, 1991.

[47] X. Shi, B. Wang, and J. K. Tsotsos, "Early recurrence improves edge detection," in *British Machine Vision Conference*, September 2013.

[48] A. L. Rothenstein, *Beyond the Limits of Feed-Forward Processing: Visual Feature Binding and Object Recognition*. PhD thesis, York University, 2011.

[49] V. Navalpakkam and L. Itti, "Search goal tunes visual features optimally," *Neuron*, vol. 53, no. 4, pp. 605–617, 2007.

[50] L. Itti and C. Koch, "Computational modelling of visual attention," *Nature Reviews Neuroscience*, vol. 2, no. 3, pp. 194–203, 2001.

[51] J. Theeuwes, "Perceptual selectivity for color and form," *Perception & Psychophysics*, vol. 51, no. 6, pp. 599–606, 1992.

[52] A. Nuthmann and J. M. Henderson, "Object-based attentional selection in scene viewing," *Journal of Vision*, vol. 10, no. 8, p. 20, 2010.

[53] H.-C. Nothdurft, "Salience from feature contrast: additivity across dimensions," *Vision Research*, vol. 40, no. 10-12, pp. 1183–1201, 2000.

[54] D. L. Robinson and S. E. Petersen, "The pulvinar and visual salience," *Trends in Neurosciences*, vol. 15, no. 4, pp. 127–132, 1992.

[55] N. P. Bichot and J. D. Schall, "Effects of similarity and history on neural mechanisms of visual selection," *Nature Neuroscience*, vol. 2, no. 6, pp. 549–554, 1999.

[56] J. Gottlieb, "Parietal mechanisms of target representation," *Current Opinion in Neurobiology*, vol. 12, no. 2, pp. 134–140, 2002.

[57] K. G. Thompson and N. P. Bichot, "A visual salience map in the primate frontal eye field," *Progress in brain research*, vol. 147, pp. 249–262, 2005.

[58] F. Murabito, C. Spampinato, S. Palazzo, D. Giordano, K. Pogorelov, and M. Riegler, "Top-down saliency detection driven by visual classification," *Computer Vision and Image Understanding*, vol. 172, pp. 67 – 76, 2018.

[59] J. H. Fecteau and D. P. Munoz, "Salience, relevance, and firing: a priority map for target selection," *Trends in Cognitive Sciences*, vol. 10, no. 8, pp. 382–390, 2006.

[60] J. Li, M. D. Levine, X. An, X. Xu, and H. He, "Visual saliency based on scale-space analysis in the frequency domain," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 4, pp. 996–1010, 2013.

[61] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint 1312.6034*, 2013.

[62] M. Kümmerer, T. S. A. Wallis, and M. Bethge, "Saliency benchmarking made easy: Separating models, maps and metrics," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[63] J. Li, C. Xia, Y. Song, S. Fang, and X. Chen, "A data-driven metric for comprehensive evaluation of saliency models," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[64] L. Itti and C. Koch, "A saliency-based search mechanism for overt and covert shifts of visual attention," *Vision Research*, vol. 40, no. 10-12, pp. 1489–1506, 2000.

[65] A. Sha'asua and S. Ullman, "Structural saliency: The detection of globally salient structures using a locally connected network," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 321–327, December 1988.

[66] B. A. Olshausen, C. H. Anderson, and D. C. V. Essen, "A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information," *The Journal of Neuroscience*, vol. 13, no. 11, pp. 4700–4719, 1993.

[67] Z. Li, "A saliency map in primary visual cortex," *Trends in Cognitive Sciences*, vol. 6, no. 1, pp. 9 – 16, 2002.

[68] V. Navalpakkam, M. Arbib, and L. Itti, "Attention and scene understanding," in *Neurobiology of Attention* (L. Itti, G. Rees, and J. K. Tsotsos, eds.), New York, NY, USA: Elsevier, 2005.

[69] Y. Zhang, E. M. Meyers, N. P. Bichot, T. Serre, T. A. Poggio, and R. Desimone, "Object decoding with attention in inferior temporal cortex," *Proceedings of the National Academy of Sciences*, vol. 108, no. 21, pp. 8850–8855, 2011.

[70] T. Buschman and S. Kastner, "From behavior to neural dynamics: An integrated theory of attention," *Neuron*, vol. 88, no. 1, pp. 127–144, 2015.

[71] M. C. Potter, "Short-term conceptual memory for pictures," *Journal of Experimental Psychology: Human Learning*, vol. 2, no. 5, pp. 509–522, 1976.

[72] M. C. Potter, B. Wyble, C. E. Hagmann, and E. S. McCourt, "Detecting meaning in RSVP at 13ms per picture," *Attention, Perception, and Psychophysics*, vol. 76, no. 2, pp. 270–279, 2014.

[73] S. Thorpe, D. Fize, and C. Marlot, "Speed of processing in the human visual system," *Nature*, vol. 381, no. 6582, pp. 520–522, 1996.

[74] J. K. Tsotsos and I. Kotseruba, "Putting saliency in its place," in *Workshop on Computational and Mathematical Models in Vision (MODVIS)*, May 2015.

[75] J. K. Tsotsos, I. Kotseruba, and C. Wloka, "Early salient region selection does not drive rapid visual categorization," *arXiv preprint 1901.04908*, 2019.

[76] M. H. Herzog and A. M. Clarke, "Why vision is not both hierarchical and feedforward," *Frontiers in Computational Neuroscience*, vol. 8, no. 135, 2014.

[77] J. M. Fuster, "Inferotemporal units in selective visual attention and short-term memory," *Journal of Neurophysiology*, vol. 64, no. 3, pp. 681–697, 1990.

[78] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Advances in Neural Information Processing Systems (NIPS)* (B. Schölkopf, J. C. Platt, and T. Hoffman, eds.), pp. 545–552, 2007.

[79] Z. Bylinskii, T. Judd, A. Borji, L. Itti, F. Durand, A. Oliva, and A. Torralba, "MIT saliency benchmark." http://saliency.mit.edu/.

[80] J. M. Findlay, "Saccade target selection during visual search," *Vision Research*, vol. 37, no. 5, pp. 617 – 631, 1997.

[81] A. Shokoufandeh, I. Marsic, and S. J. Dickinson, "View-based object recognition using saliency maps," *Image and Vision Computing*, vol. 17, no. 5-6, pp. 445 – 460, 1999.

[82] M. Bar, "A cortical mechanism for triggering top-down facilitation in visual object recognition," *Journal of Cognitive Neuroscience*, vol. 15, no. 4, pp. 600–609, 2003.

[83] A. Torralba, "Modeling global scene factors in attention," *Journal of the Optical Society of America*, vol. 20, no. 7, pp. 1407–1418, 2003.

[84] A. Oliva, A. Torralba, M. S. Castelhano, and J. M. Henderson, "Top-down control of visual attention in object detection," in *International Conference on Image Processing (ICIP)*, vol. 1, pp. I–253–6, September 2003.

[85] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[86] R. A. Rensink, "Scene perception," in *Encyclopedia of Psychology* (A. Kazdin, ed.), vol. 7, pp. 151–155, New York, NY, USA: Oxford University Press, 2000.

[87] A. Oliva, "Gist of the scene," *Neurobiology of Attention*, vol. 696, no. 64, pp. 251–258, 2005.

[88] A. Oliva and A. Torralba, "Building the gist of a scene: the role of global image features in recognition," in *Visual Perception Fundamentals of Awareness: Multi-Sensory Integration and High-Order Perception* (S. Martinez-Conde, S. Macknik, L. Martinez, J.-M. Alonso, and P. Tse, eds.), vol. 155, Part B of *Progress in Brain Research*, pp. 23 – 36, Elsevier, 2006.

[89] E. Fazl-Ersi and J. K. Tsotsos, "Histogram of oriented uniform patterns for robust place recognition and categorization," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 468–483, 2012.

[90] D. Walther and C. Koch, "Modeling attention to salient proto-objects," *Neural Networks*, vol. 19, no. 9, pp. 1395 – 1407, 2006. Special Issue on Brain and Attention.

[91] W. Einhäuser, M. Spain, and P. Perona, "Objects predict fixations better than early saliency," *Journal of Vision*, vol. 8, no. 14, p. 18, 2008.

[92] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 73–80, June 2010.

[93] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai, "Fusing generic objectness and visual saliency for salient object detection," in *IEEE International Conference on Computer Vision (ICCV)*, November 2011.

[94] A. Borji, D. N. Sihite, and L. Itti, "Objects do not predict fixations better than early saliency: A re-analysis of Einhäuser et al.'s data," *Journal of Vision*, vol. 13, no. 10, p. 18, 2013.

[95] B. W. Tatler, R. J. Baddeley, and I. D. Gilchrist, "Visual correlates of fixation selection: effects of scale and time," *Vision Research*, vol. 45, no. 5, pp. 643–659, 2005.

[96] D. Parkhurst, K. Law, and E. Niebur, "Modeling the role of salience in the allocation of overt visual attention," *Vision Research*, vol. 42, no. 1, pp. 107–123, 2002.

[97] J. K. Tsotsos, I. Kotseruba, and C. Wloka, "A focus on selection for fixation," *Journal of Eye Movement Research*, vol. 9, no. 5, pp. 1–34, 2016.

[98] J. K. Tsotsos and W. Kruijne, "Cognitive programs: Software for attention's executive," *Frontiers in Psychology*, vol. 5, no. 1260, 2014.

[99] S. Ullman, "Visual routines," *Cognition*, vol. 18, no. 1-3, pp. 97–159, 1984.

[100] C. Koch and N. Tsuchiya, "Attention and consciousness: two distinct brain processes," *Trends in Cognitive Sciences*, vol. 11, no. 1, pp. 16–22, 2007.

[101] H.-C. Nothdurft, "Saliency effects across dimensions in visual search," *Vision Research*, vol. 33, no. 5, pp. 839 – 844, 1993.

[102] M.-S. Kim and K. R. Cave, "Top-down and bottom-up attentional control: On the nature of interference from a salient distractor," *Perception & Psychophysics*, vol. 61, no. 6, pp. 1009–1023, 1999.

[103] J. Theeuwes, "Top-down search strategies cannot override attentional capture," *Psychonomic Bulletin & Review*, vol. 11, no. 1, pp. 65–70, 2004.

[104] D. Lamy and Y. Tsal, "A salient distractor does not disrupt conjunction search," *Psychonomic Bulletin & Review*, vol. 6, pp. 93–98, Mar 1999.

[105] G. J. Zelinsky, W. Zhang, B. Yu, X. Chen, and D. Samaras, "The role of top-down and bottom-up processes in guiding eye movements during visual search," in *Advances in Neural Information Processing Systems (NIPS)* (Y. Weiss, B. Schölkopf, and J. C. Platt, eds.), pp. 1569–1576, 2006.

[106] G. J. Zelinsky, "A theory of eye movements during target acquisition," *Psychological Review*, vol. 115, no. 4, pp. 787–835, 2008.

[107] F. H. Hamker, "Modeling feature-based attention as an active top-down inference process," *Biosystems*, vol. 86, no. 1, pp. 91 – 99, 2006. Special Issue Brain, Vision and Artificial Intelligence (BVAi) 2005: Papers from a Symposium on Brain Basics and Natural Vision.

[108] H. Yang and G. J. Zelinsky, "Visual search is guided to categorically-defined targets," *Vision Research*, vol. 49, no. 16, pp. 2095 – 2103, 2009.

[109] M. B. Neider and G. J. Zelinsky, "Scene context guides eye movements during visual search," *Vision Research*, vol. 46, no. 5, pp. 614 – 621, 2006.

[110] V. Maljkovic and K. Nakayama, "Priming of pop-out: I. role of features," *Memory and Cognition*, vol. 22, no. 6, pp. 657–672, 1994.

[111] V. Maljkovic and K. Nakayama, "Priming of pop-out: Ii. the role of position," *Perception & Psychophysics*, vol. 58, no. 7, pp. 977–991, 1996.

[112] J. M. Henderson, G. L. Malcolm, and C. Schandl, "Searching in the dark: Cognitive relevance drives attention in real-world scenes," *Psychonomic Bulletin & Review*, vol. 16, no. 5, 2009.

[113] T. Foulsham and G. Underwood, "How does the purpose of inspection influence the potency of visual salience in scene perception?," *Perception*, vol. 36, no. 8, pp. 1123–1138, 2007.

[114] T. Foulsham and A. Kingstone, "Modelling the influence of central and peripheral information on saccade biases in gaze-contingent scene viewing," *Visual Cognition*, vol. 20, no. 4-5, pp. 546–579, 2012.

[115] J. Najemnik and W. S. Geisler, "Optimal eye movmement strategies in visual search," *Nature*, vol. 434, no. 7031, pp. 387–391, 2005.

[116] L. Johnson, B. Sullivan, M. Hayhoe, and D. Ballard, "Predicting human visuomotor behaviour in a driving task," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 369, no. 1636, p. 20130044, 2014.

[117] H.-k. Ko, M. Poletti, and M. Rucci, "Microsaccades precisely relocate gaze in a high visual acuity task," *Nature Neuroscience*, vol. 13, no. 12, pp. 1549–1553, 2010.

[118] R. M. Steinman, Z. Pizlo, T. I. Forofonova, and J. Epelboim, "One fixates accurately in order to see clearly not because one sees clearly," *Spatial vision*, vol. 16, no. 3, pp. 225–241, 2003.

[119] S. Yantis, *Visual Search*. East Sussex, UK: Psychology Press, 1998.

[120] S. Marat, T. HoPhuoc, L. Granjon, N. Guyader, D. Pellerin, and A. Guérin-Dugué, "Modelling spatio-temporal saliency to predict gaze direction for short videos," *International Journal of Computer Vision*, vol. 82, no. 3, pp. 231–243, 2009.

[121] V. Mahadevan and N. Vasconcelos, "Spatiotemporal saliency in dynamic scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 171–177, Jan 2010.

[122] A. Zaharescu and R. P. Wildes, "Spatiotemporal salience via centre-surround comparison of visual spacetime orientations," in *Asian Conference on Computer Vision (ACCV)*, November 2012.

[123] A. P. Hillstrom and S. Yantis, "Visual motion and attentional capture," *Perception & Psychophysics*, vol. 55, no. 4, pp. 399–411, 1994.

[124] T. J. Smith and P. K. Mital, "Attentional synchrony and the influence of viewing task on gaze behavior in static and dynamic scenes," *Journal of Vision*, vol. 13, no. 8, p. 16, 2013.

[125] A. Borji, D. N. Sihite, and L. Itti, "Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 55–69, 2013.

[126] N. Riche, M. Mancas, D. Culibrk, V. Crnojevic, B. Gosselin, and T. Dutoit, "Dynamic saliency models and human attention: a comparative study on videos," in *Asian Conference on Computer Vision (ACCV)*, November 2012.

[127] L. C. Loschky, A. M. Larson, J. P. Magliano, and T. J. Smith, "What would jaws do? The tyranny of film and the relationship between gaze and higher-level narrative film comprehension," *PLOS ONE*, vol. 10, pp. 1–23, 11 2015.

[128] D. M. Chandler, "Seven challenges in image quality assessment: Past, present, and future research," *ISRN Signal Processing*, pp. 1–53, 2013.

[129] Q. Ma and L. Zhang, "Saliency-based image quality assessment criterion," in *International Conference on Intelligent Computing (ICIC)*, pp. 1124–1133, September 2008.

[130] T. Yubing, H. Konik, F. A. Cheikh, and A. Tremeau, "Full reference image quality assessment based on saliency map analysis," *Journal of Imaging Science and Technology*, vol. 54, no. 3, pp. 1–15, 2010.

[131] J. Y. Lin, T. J. Liu, W. Lin, and C. C. J. Kuo, "Visual-saliency-enhanced image quality assessment indices," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1–4, October 2013.

[132] N. Dhavale and L. Itti, "Saliency-based multifoveated MPEG compression," in *International Symposium on Signal Processing and Its Applications*, vol. 1, pp. 229–232, July 2003.

[133] S. X. Yu and D. A. Lisin, "Image compression based on visual saliency at individual scales," in *International Symposium on Advances in Visual Computing (ISVC)*, pp. 157–166, November 2009.

[134] "Global internet phenomena: Africa, middle east, and north america," tech. rep., Sandvine, 2015.

[135] P. Harding and N. Roberston, "Task-based visual saliency for intelligent compression," in *IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, November 2009.

[136] K. Shubina and J. K. Tsotsos, "Visual search for an object in a 3D environment using a mobile robot," tech. rep., York University, 2008.

[137] F. Saidi, O. Stasse, and K. Yokoi, *Active Visual Search by a Humanoid Robot*, pp. 171–184. Springer Berlin Heidelberg, 2008.

[138] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Dynamically encoded actions based on space-time saliency," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[139] V. Yanulevskaya, J. Uijlings, J.-M. Geusebroek, N. Sebe, and A. Smeulders, "A proto-object-based computational model for visual saliency," *Journal of Vision*, vol. 13, no. 13, p. 27, 2013.

[140] A. F. Russell, S. Mihalas, R. von der Heydt, E. Niebur, and R. Etienne-Cummings, "A model of proto-object based saliency," *Vision Research*, vol. 94, pp. 1 – 15, 2014.

[141] J. Harel, "A saliency implementation in MATLAB." `http://http://www.klab.caltech.edu/~harel/share/gbvs.php`.

[142] A. Derrington, J. Krauskopf, and P. Lennie, "Chromatic mechanisms in lateral geniculate nucleus of macaque," *Journal of Physiology*, vol. 357, no. 1, pp. 241–65, 1984.

[143] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 1915–1926, 2012.

[144] S. J. Sangwine and R. E. Horne, eds., *The colour image processing handbook*. Springer Science & Business Media, 2012.

[145] E. Erdem and A. Erdem, "Visual saliency estimation by nonlinearly integrating features using region covariances.," *Journal of Vision*, vol. 13, no. 2013, p. 11, 2013.

[146] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *European Conference on Computer Vision (ECCV)*, pp. 589–600, May 2006.

[147] N. Riche, M. Mancas, B. Gosselin, and T. Dutoit, "RARE: A new bottom-up saliency model," in *IEEE International Conference on Image Processing (ICIP)*, pp. 641–644, September 2012.

[148] N. Riche, M. Mancas, M. Duvinage, M. Mibulumukini, B. Gosselin, and T. Dutoit, "RARE2012: A multi-scale rarity-based saliency detection with its comparative statistical analysis," *Signal Processing: Image Communication*, vol. 28, no. 6, pp. 642–658, 2013.

[149] J. Leroy, N. Riche, M. Mancas, B. Gosselin, and T. Dutoit, "SuperRare: an object-oriented saliency algorithm based on superpixels rarity," *IEEE International Conference on Robotics and Automation (ICRA)*, May 2014.

[150] S. Frintrop, T. Werner, and G. M. Garca, "Traditional saliency reloaded: A good old model in new shape," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[151] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[152] J. Zhang and S. Sclaroff, "Exploiting surroundedness for saliency detection: A boolean map approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, pp. 889–902, May 2016.

[153] N. D. B. Bruce, X. Shi, E. Simine, and J. K. Tsotsos, "Visual representation in the determination of saliency," in *Canadian Conference on Computer and Robot Vision (CRV)*, May 2011.

[154] N. D. B. Bruce and J. K. Tsotsos, "An information theoretic model of saliency and visual search," in *International Workshop on Attention and Performance in Computer Vision (WAPCV)* (E. R. L. Paletta, ed.), pp. 171–183, 2007.

[155] N. D. B. Bruce and J. K. Tsotsos, "Saliency, attention, and visual search: An information theoretic approach," *Journal of Vision*, vol. 9, no. 3, pp. 1–24, 2009.

[156] N. Bruce and J. Tsotsos, "Visual representation determines search difficulty: Explaining visual search asymmetries," *Frontiers in Computational Neuroscience*, vol. 5, no. 33, 2011.

[157] N. D. B. Bruce, C. Wloka, N. Frosst, S. Rahman, and J. K. Tsotsos, "On computational modeling of visual saliency: Examining what's right, and what's left," *Vision Research*, vol. 116, pp. 95 – 112, 2015. Special Issue on Computational Models of Visual Attention.

[158] A. Hyvarinen, P. Hoyer, and M. Inki, "Topographic ICA as a model of V1 receptive fields," in *IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN)*, vol. 4, pp. 83–88, 2000.

[159] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, "SUN: A Bayesian framework for saliency using natural statistics," *Journal of Vision*, vol. 8, no. 7:32, pp. 1–20, 2008.

[160] X. Hou and L. Zhang, "Dynamic visual attention: Searcing for coding length increments," in *Advances in Neural Information Processing Systems (NIPS)* (D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, eds.), pp. 681–688, 2009.

[161] H. J. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance.," *Journal of Vision*, vol. 9, no. 12, pp. 1–27, 2009.

[162] A. Garcia-Diaz, X. R. Fdez-Vidal, X. M. Pardo, and R. Dosil, "Saliency from hierarchical adaptation through decorrelation and variance normalization," *Image and Vision Computing*, vol. 30, no. 1, pp. 51–64, 2012.

[163] H. B. Barlow, *Possible principles underlying the transformation of sensory messages sensory communication*. Cambridge, MA, USA: MIT Press, 1961.

[164] H. R.-Tavakoli, A. Atyabi, A. Rantanen, S. J. Laukka, S. Nefti-Meziani, and J. Heikkilä, "Predicting the valence of a scene from observers eye movements," *PLoS ONE*, vol. 10, pp. 1–19, 09 2015.

[165] S. Fang, J. Li, Y. Tian, T. Huang, and X. Chen, "Learning discriminative subspaces on random contrasts for image saliency analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 5, pp. 1095–1108, 2017.

[166] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2007.

[167] B. Schauerte and R. Stiefelhagen, "Quaternion-based spectral saliency detection for eye fixation prediction," in *European Conference on Computer Vision (ECCV)*, October 2012.

[168] X. Hou, J. Harel, and C. Koch, "Image signature: Highlighting sparse salient regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 194–201, 2012.

[169] A. Borji, H. R. Tavakoli, D. N. Sihite, and L. Itti, "Analysis of scores, datasets, and models in visual saliency prediction," in *IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[170] B. W. Tatler and B. T. Vincent, "The prominence of behavioural biases in eye guidance," *Visual Cognition*, vol. 17, no. 6-7, pp. 1029–1054, 2009.

[171] O. Le Meur and A. Coutrot, "Introducing context-dependent and spatially-variant viewing biases in saccadic models," *Vision Research*, vol. 121, pp. 72–84, 2016.

[172] R. Rosenholtz, "A simple saliency model predicts a number of motion popout phenomena," *Vision Research*, vol. 39, no. 19, pp. 3157 – 3163, 1999.

[173] P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[174] P. Felzenswalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2008.

[175] M. Kümmerer, L. Theis, and M. Bethge, "Deep Gaze I: Boosting saliency prediction with feature maps trained on ImageNet," in *International Conference on Learning Representations (ICLR) Workshop*, May 2015.

[176] S. Barthelmé, H. Trukenbrod, R. Engbert, and F. Wichmann, "Modeling fixation locations using spatial point processes," *Journal of Vision*, vol. 13, no. 12, pp. 1–34, 2013.

[177] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *International Conference on Machine Learning (ICML)*, June 2014.

[178] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)* (F. Pereira, C. Burges, L. Bottou, and K. Weinberger, eds.), pp. 1097–1105, 2012.

[179] N. Liu, J. Han, D. Zhang, S. Wen, and T. Liu, "Predicting eye fixations using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[180] S. S. Kruthiventi, K. Ayush, and R. V. Babu, "DeepFix: A fully convolutional neural network for predicting human eye fixations," *arXiv preprint 1510.02927*, 2015.

[181] M. Jiang, S. Huang, J. Duan, and Q. Zhao, "SALICON: Saliency in context," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[182] M. Kümmerer, T. S. A. Wallis, and M. Bethge, "Deepgaze II: reading fixations from deep features trained on object recognition," *arXiv preprint 1610.01563*, 2016.

[183] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint 1409.1556*, 2014.

[184] M. Kümmerer, T. S. A. Wallis, L. A. Gatys, and M. Bethge, "Understanding low- and high-level contributions to fixation prediction," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2017.

[185] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," in *Matters of Intelligence*, pp. 115–141, Dordrecht, NED: Springer, 1987.

[186] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, "Predicting human eye fixations via an lstm-based saliency attentive model," *arXiv preprint 1611.09571*, 2016.

[187] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint 1512.03385*, 2015.

[188] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, "A deep multi-level network for saliency prediction," in *International Conference on Pattern Recognition (ICPR)*, pp. 3488–3493, December 2016.

[189] J. Pan, C. Canton-Ferrer, K. McGuinness, N. E. O'Connor, J. Torres, E. Sayrol, and X. Giró i Nieto, "Salgan: Visual saliency prediction with generative adversarial networks," *arXiv preprint 1701.01081*, 2017.

[190] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems (NIPS)* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), 2014.

[191] W. Wang and J. Shen, "Deep visual attention prediction," *Transactions on Image Processing*, vol. 27, no. 5, pp. 2368–2378, 2018.

[192] Z. Bylinskii, E. M. DeGennaro, R. Rajalingham, H. Ruda, J. Zhang, and J. Tsotsos, "Towards the quantitative evaluation of visual attention models," *Vision Research*, vol. 116, pp. 258 – 268, 2015. Special Issue on Computational Models of Visual Attention.

[193] B. W. Tatler, "The central fixation bias in scene viewing: Selecting an optimal viewing position independently of motor biases and image feature distributions," *Journal of Vision*, vol. 7, no. 14, pp. 1–17, 2007.

[194] C. Wloka, "Integrating overt and covert attention using peripheral and central processing streams," Master's thesis, York University, Toronto, ON, CA, 2012.

[195] A. Borji and L. Itti, "Defending yarbus: Eye movements reveal observers' task," *Journal of Vision*, vol. 14, no. 3, p. 29, 2014.

[196] S. Frintrop, G. Backer, and E. Rome, "Goal-directed search with a top-down modulated computational attention system," in *Pattern Recognition*, vol. 3663, pp. 117–124, Berlin, Germany: Springer, 2005.

[197] C. Kanan, M. H. Tong, L. Zhang, and G. W. Cottrell, "SUN: Top-down saliency using natural statistics," *Visual Cognition*, vol. 17, no. 6-7, pp. 979–1003, 2009.

[198] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look." `http://people.csail.mit.edu/tjudd/WherePeopleLook/`.

[199] B. Russel, A. Torralba, K. Murphy, and W. T. Freeman, "LabelMe: a database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, no. 1-3, 2007.

[200] I. van der Linde, U. Rajashekar, A. C. Bovik, and L. K. Cormack, "DOVES: A database of visual eye movements." `http://live.ece.utexas.edu/research/doves`.

[201] J. H. van Hateren and A. van der Schaaf, "Independent component filters of natural images compared with simple cells in primary visual cortex," *Proceedings: Biological Sciences*, vol. 265, no. 1394, pp. 359–366, 1998.

[202] A. N. Borodin and I. A. Ibragimov, "Limit theorems for functionals of random walks," in *Proceedings of the Steklov Institute of Mathematics* (V. N. Sudakov, ed.), vol. 195, 1995.

[203] D. Brockmann and T. Geisel, "The ecology of gaze shifts," *Neurocomputing*, vol. 32-33, pp. 643–650, 2000.

[204] G. Boccignone and M. Ferraro, "Modelling gaze shift as a constrained random walk," *Physica A: Statistical Mechanics and its Applications*, vol. 331, no. 1-2, pp. 207 – 218, 2004.

[205] R. J. Peters, A. Iyer, L. Itti, and C. Koch, "Components of bottom-up gaze allocation in natural images," *Vision Research*, vol. 45, no. 18, pp. 2397–2416, 2005.

[206] A. Borji, D. N. Sihite, and L. Itti, "Salient object detection: A benchmark," in *European Conference on Computer Vision (ECCV)*, October 2012.

[207] M. A. Islam, M. Kalash, and N. D. Bruce, "Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of multiple salient objects," *arXiv preprint 1803.05082*, 2018.

[208] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 59–66, January 1998.

[209] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[210] O. Le Meur and T. Baccino, "Methods for comparing scanpaths and saliency maps: strengths and weaknesses," *Behavior Research Methods*, vol. 45, no. 1, pp. 251–266, 2012.

[211] L. Itti and P. F. Baldi, "Bayesian surprise attracts human attention," in *Advances in neural information processing systems (NIPS)* (Y. Weiss, B. Schölkopf, and J. C. Platt, eds.), pp. 547–554, 2006.

[212] D. M. Green, *Signal detection theory and psychophysics*, vol. 1. New York, NY, USA: Wiley, 1966.

[213] D. Parkhurst and E. Niebur, "Scene content selected by active vision," *Spatial Vision*, vol. 16, no. 2, pp. 125–154, 2003.

[214] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," 2009.

[215] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond PASCAL: A benchmark for 3D object detection in the wild," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2014.

[216] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[217] W. Zhang, Y. Tian, X. Zha, and H. Liu, "Benchmarking state-of-the-art visual saliency models for image quality assessment," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2016.

[218] J. Nielsen, *F-Shaped Pattern for Reading Web Content*. Nielsen Norman Group, 2006.

[219] J.-F. Tsai and K.-J. Chang, "OpenSource implementation of context-aware saliency detection." `https://sites.google.com/a/jyunfan.co.cc/site/opensource-1/contextsaliency`.

[220] C. L. Thomas, "OpenSalicon: An open source implementation of the Salicon saliency model," Tech. Rep. TR-2016-02, University of Pittsburgh, 2016.

[221] N. M. W. Frosst, C. Wloka, and J. K. Tsotsos, "The effects of image padding in saliency algorithms," *Perception*, vol. 43 ECVP Abstract Supplement, p. 106, 2014.

[222] C. Goble, "Better software, better research," *IEEE Internet Computing*, vol. 18, pp. 4–8, September 2014.

[223] Y. Alnoamany and J. A. Borghi, "Towards computational reproducibility: researcher perspectives on the use and sharing of software," *PeerJ Computer Science*, vol. 4, p. e163, 2018.

[224] D. Berga and X. Otazu, "A neurodynamic model of saliency prediction in V1," *arXiv preprint 1811.06308*, 2018.

[225] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint 1408.5093*, 2014.

[226] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from tensorflow.org.

[227] H. R. Tavakoli, E. Rahtu, and J. Heikkilä, "Fast and efficient saliency detection using sparse sampling and kernel density estimation.," in *Proceedings of Scandinavian Conference on Image Analysis (SCIA)*, 2011.

[228] J. Han, D. Zhang, G. Cheng, N. Liu, and D. Xu, "Advanced deep-learning techniques for salient and category-specific object detection: A survey," *IEEE Signal Processing Magazine*, vol. 35, pp. 84–100, January 2018.

[229] M. Mancas, D. Unay, B. Gosselin, and B. Macq, "Computational attention for defect localization," in *Proc. of ICVS Workshop on Computational Attention and Applications*, 2007.

[230] O. Boiman and M. Irani, "Detecting irregularities in image and in video," *International Journal of Computer Vision*, vol. 74, no. 1, pp. 17–31, 2007.

[231] B. Suh, H. Ling, B. B. Bederson, and D. W. Jacobs, "Automatic thumbnail cropping and its effectiveness," in *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (UIST)*, pp. 95–104, 2003.

[232] T. Zhu, W. Wang, P. Liu, and Y. Xie, "Saliency-based adaptive scaling for image retargeting," in *2011 Seventh International Conference on Computational Intelligence and Security*, pp. 1201–1205, December 2011.

[233] C. M. Masciocchi and J. D. Still, "Alternatives to eye tracking for predicting stimulus-driven attentional selection within interfaces," *HumanComputer Interaction*, vol. 28, no. 5, pp. 417–441, 2013.

[234] K. Fu, J. Li, H. Shen, and Y. Tian, "How drones look: Crowdsourced knowledge transfer for aerial video saliency prediction," *arXiv preprint 1811.05625*, 2018.

[235] B. J. White, D. J. Berg, J. Y. Kan, R. A. Marino, L. Itti, and D. P. Munoz, "Superior colliculus neurons encode a visual saliency map during free viewing of natural dynamic video," *Nature Communications*, vol. 8, p. 14263, Jan. 2017.

[236] D. H. Brainard, "The psychophysics toolbox," *Spatial Vision*, vol. 10, pp. 433–436, 1997.

[237] Q. Fang, "jsonlab toolbox." `https://github.com/fangq/jsonlab`.

[238] Docker, Inc., "Docker Community Edition." `https://github.com/docker/docker-ce`.

[239] NVIDIA, "NVIDIA Container Runtime for Docker." `https://github.com/NVIDIA/nvidia-docker`.

[240] Docker, Inc., "Docker Hub." `https://www.docker.com/products/docker-hub`.

[241] T. Töllner, M. Zehetleitner, K. Gramann, and H. J. Müller, "Stimulus saliency modulates pre-attentive processing speed in human visual cortex," *PLoS ONE*, vol. 6, pp. 1–8, 01 2011.

[242] B. C. Motter and E. J. Belky, "The guidance of eye movements during active visual search," *Vision Research*, vol. 38, no. 12, pp. 1805–1815, 1998.

[243] M. F. Raymond M. Klein, "Search performance without eye movements," *Perception & Psychophysics*, vol. 46, pp. 476–482, 1989.

[244] J. Duncan and G. W. Humphreys, "Visual search and stimulus similarity.," *Psychological review*, vol. 96, no. 3, p. 433, 1989.

[245] M. Zehetleitner, A. I. Koch, H. Goschy, and H. J. Mller, "Salience-based selection: Attentional capture by distractors less salient than the target," *PLoS ONE*, vol. 8, pp. 1–14, 01 2013.

[246] C. Wloka, S.-A. Yoo, R. Sengupta, T. Kunić, and J. K. Tsotsos, "Psychophysical evaluation of saliency algorithms," *Journal of Vision*, vol. 16, no. 12, p. 1291, 2016.

[247] J. Kim, M. Ricci, and T. Serre, "Not-So-CLEVR: learning same–different relations strains feedforward neural networks," *Interface focus*, vol. 8, no. 4, pp. 1–13, 2018.

[248] A. Treisman and J. Souther, "Search asymmetry: A diagnostic for preattentive processing of separable features," *Journal of Experimental Psychology: General*, vol. 114, no. 3, pp. 285–310, 1985.

[249] A. Treisman and S. Gormican, "Feature analysis in early vision: Evidence from search asymmetries," *Psychological Review*, vol. 95, no. 1, pp. 15–48, 1988.

[250] J. Saiki, T. Koike, K. Takahashi, and T. Inoue, "Visual search asymmetry with uncertain targets," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 31, no. 6, pp. 1274–1287, 2005.

[251] R. Rosenholtz, "Search asymmetries? What search asymmetries?," *Perception & Psychophysics*, vol. 63, pp. 476–489, April 2001.

[252] R. Rosenholtz, A. L. Nagy, and N. R. Bell, "The effect of background color on asymmetries in color search," *Journal of Vision*, vol. 4, no. 3, p. 9, 2004.

[253] J. M. Wolfe, "Asymmetries in visual search: An introduction," *Perception & Psychophysics*, vol. 63, pp. 381–389, April 2001.

[254] J. Shen and E. M. Reingold, "Visual search asymmetry: The influence of stimulus familiarity and low-level features," *Perception & Psychophysics*, vol. 63, no. 3, pp. 464–475, 2001.

[255] J. M. Wolfe and T. S. Horowitz, "What attributes guide the deployment of visual attention and how do they do it?," *Nature Reviews Neuroscience*, vol. 5, no. 6, pp. 495–501, 2004.

[256] J. M. Wolfe and T. S. Horowitz, "Five factors that guide attention in visual search," *Nature Human Behaviour*, vol. 1, no. 3, 2017.

[257] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, p. 607, 1996.

[258] R. Rauschenberger and S. Yantis, "Perceptual encoding efficiency in visual search," *Journal of Experimental Psychology*, vol. 135, no. 1, pp. 116–131, 2006.

[259] M. Kümmerer, T. Wallis, and M. Bethge, "How close are we to understanding image-based saliency?," *arXiv preprint 1409.7686*, 2014.

[260] M. Kümmerer, T. S. A. Wallis, and M. Bethge, "Information-theoretic model comparison unifies saliency metrics," *Proceedings of the National Academy of Sciences*, vol. 112, no. 52, pp. 16054–16059, 2015.

[261] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *arXiv preprint 1405.0312*, 2014.

[262] G. Horstmann and A. Bauland, "Search asymmetries with real faces: Testing the anger-superiority effect," *Emotion*, vol. 6, no. 2, pp. 193–207, 2006.

[263] C. Ballaz, L. Boutsen, C. Peyrin, G. W. Humphreys, and C. Marendaz, "Visual search for object orientation can be modulated by canonical orientation," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 31, no. 1, pp. 20–39, 2005.

[264] S. I. Becker, "The mechanism of priming: Episodic retrieval or priming of pop-out?," *Acta Psychologica*, vol. 127, no. 2, pp. 324 – 339, 2008.

[265] G. Porter, A. Tales, T. Troscianko, G. Wilcock, J. Haworth, and U. Leonards, "New insights into feature and conjunction search: I. evidence from pupil size, eye movements and ageing," *Cortex*, vol. 46, no. 5, pp. 621 – 636, 2010.

[266] B. Welch, "The Generalization of 'Student's Problem When Several Different Population Variances are Involved," *Biometrika*, vol. 34, pp. 28–35, 01 1947.

[267] P. Baldi and L. Itti, "Of bits and wows: A bayesian theory of surprise with applications to attention," *Neural Networks*, vol. 23, no. 5, pp. 649 – 666, 2010.

[268] M. Ricci, J. Kim, and T. Serre, "Not-So-CLEVR: Visual relations strain feedforward neural networks," *arXiv preprint 1802.03390*, 2018.

[269] G. Underwood, T. Foulsham, E. Loon, L. Humphreys, and J. Bloyce, "Eye movements during scene inspection: A test of the saliency map hypothesis," vol. 18, pp. 321–342, 05 2006.

[270] N. C. Anderson, E. Ort, W. Kruijne, M. Meeter, and M. Donk, "It depends on when you look at it: Salience influences eye movements in natural scene viewing and search early in time," *Journal of Vision*, vol. 15, no. 5, p. 9, 2015.

[271] J. P. D. Vries, S. V. der Stigchel, and I. T. C. Hooge, "The lifetime of salience extends beyond the initial saccade," *Perception*, vol. 47, no. 2, pp. 125–142, 2017.

[272] P. Besse, B. Guillouet, J.-M. Loubes, and F. Royer, "Review and perspective for distance based clustering of vehicle trajectories," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 11, pp. 3306–3317, 2016.

[273] S. Mannan, K. Ruddock, and D. Wooding, "Fixation sequences made during visual examination of briefly presented 2d images," *Spatial Vision*, vol. 11, no. 2, pp. 157–178, 1997.

[274] H. Strasburger, I. Rentschler, and M. Jüttner, "Peripheral vision and pattern recognition: A review," *Journal of Vision*, vol. 11, no. 5, pp. 1–82, 2011.

[275] X. Kuang, M. Poletti, J. D. Victor, and M. Rucci, "Temporal encoding of spatial information during active visual fixation," *Current Biology*, vol. 22, no. 6, pp. 510 – 514, 2012.

[276] J. K. Tsotsos, "On the relative complexity of active vs. passive visual search," *International Journal of Computer Vision*, vol. 7, no. 2, pp. 127–141, 1992.

[277] R. Bajcsy, Y. Aloimonos, and J. K. Tsotsos, "Revisiting active perception," *Autonomous Robots*, vol. 42, no. 2, pp. 177–196, 2017.

[278] J. Pola and H. J. Wyatt, "Smooth pursuit: response characteristics, stimuli and mechanisms," in *Eye Movements* (R. Carpenter, ed.), pp. 138–157, CRC Press, 1991.

[279] C. Distler and K.-P. Hoffmann, "The optokinetic reflex," in *The Oxford Handbook of Eye Movements* (S. P. Liversedge, I. D. Gilchrist, and S. Everling, eds.), ch. 4, pp. 65–83, Oxford, UK: Oxford University Press, 2011.

[280] J. Crawford and E. Klier, "Neural control of three-dimensional gaze shifts," in *The Oxford Handbook of Eye Movements* (S. P. Liversedge, I. D. Gilchrist, and S. Everling, eds.), ch. 18, pp. 339–356, Oxford, UK: Oxford University Press, 2011.

[281] B. White and D. Munoz, "The superior colliculus," in *The Oxford Handbook of Eye Movements* (S. P. Liversedge, I. D. Gilchrist, and S. Everling, eds.), ch. 11, pp. 195–214, Oxford, UK: Oxford University Press, 2011.

[282] F. Miles, "The cerebellum," in *Eye Movements* (R. Carpenter, ed.), pp. 224–243, CRC Press, 1991.

[283] C. Vokoun, S. Mahamed, and M. Basso, "Saccadic eye movements and the basal ganglia," in *The Oxford Handbook of Eye Movements* (S. P. Liversedge, I. D. Gilchrist, and S. Everling, eds.), ch. 12, pp. 215–234, Oxford, UK: Oxford University Press, 2011.

[284] D. Noton and L. Stark, "Scanpaths in eye movements during pattern perception," *Science*, vol. 171, no. 3968, pp. 308–311, 1971.

[285] T. Foulsham and A. Kingstone, "Fixation-dependent memory for natural scenes: An experimental test of scanpath theory," *Journal of Experimental Psychology: General*, vol. 142, no. 1, pp. 41–56, 2013.

[286] 3M Visual Attention Service, *3M White Van VAS Sample Report*, 2015. Version 5.2.

[287] 3M Commercial Graphics Division, *3M Visual Attention Service Validation Study*, 2010.

[288] J. Harold, I. Lorenzoni, T. F. Shipley, and K. R. Coventry, "Cognitive and psychological science insights to improve climate change data visualization," *Nature Climate Change*, vol. 6, no. 12, pp. 1080–1089, 2016.

[289] J. Gaspar, N. Winters, and J. Santos-Victor, "Vision-based navigation and environmental representations with an omnidirectional camera," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 890–898, 2000.

[290] J. Elder, Y. Hou, R. Goldstein, and F. Dornaika, "Attentive panoramic visual sensor," Oct 31 2006. US Patent 7,130,490.

[291] N. Mavridis, *Grounded situation models for situated conversational assistants*. PhD thesis, Massachusetts Institute of Technology, 2007.

[292] Z. Ycel, A. Salah, C. Mericli, T. Mericli, R. Valenti, and T. Gevers, "Joint attention by gaze interpolation and saliency," *IEEE Transactions on Cybernetics*, vol. 43, no. 3, pp. 829–842, 2013.

[293] I. Kotseruba, A. Rasouli, and J. K. Tsotsos, "Joint attention in autonomous driving (JAAD)," *arXiv preprint 1609.04741*, 2016.

[294] A. Moon, D. M. Troniak, B. Gleeson, M. K. Pan, M. Zheng, B. A. Blumer, K. MacLean, and E. A. Croft, "Meet me where I'm gazing: How shared attention gaze affects human-robot handover timing," in *Proceedings of the ACM/IEEE International Conference on Human-robot Interaction*, 2014.

[295] B. B. Mandelbrot, *The fractal geometry of nature*. WH Freeman New York, 1983.

[296] S. Mathe and C. Sminchisescu, "Action from still image dataset and inverse optimal control to learn task specific visual scanpaths," in *Advances in Neural Information Processing Systems (NIPS)* (C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), pp. 1923–1931, 2013.

[297] H. Liu, D. Xu, Q. Huang, W. Li, M. Xu, and S. Lin, "Semantically-based human scanpath estimation with hmms," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 3232–3239, December 2013.

[298] M. Jiang, X. Boix, G. Roig, J. Xu, L. Van Gool, and Q. Zhao, "Learning to predict sequences of human visual fixations," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, pp. 1241–1252, June 2016.

[299] T. Ngo and B. S. Manjunath, "Saccade gaze prediction using a recurrent neural network," in *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 3435–3439, September 2017.

[300] Z. Chen and W. Sun, "Scanpath prediction for visual attention using IOR-ROI LSTM," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pp. 642–648, International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[301] M. Assens, X. G. i Nieto, K. McGuinness, and N. E. O'Connor, "PathGAN: Visual scanpath prediction with generative adversarial networks," 2018.

[302] X. Sun, H. Yao, and R. Ji, "What are we looking for: Towards statistical modeling of saccadic eye movements and visual saliency," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1552–1559, June 2012.

[303] A. Rosenfeld, M. Biparva, and J. K. Tsotsos, "Priming neural networks," *arXiv preprint 1711.05918*, 2017.

[304] P. Gable and E. Harmon-Jones, "The motivational dimensional model of affect: Implications for breadth of attention, memory, and cognitive categorisation," *Cognition and Emotion*, vol. 24, no. 2, pp. 322–337, 2010.

[305] Y. Niu, R. M. Todd, M. Kyan, and A. K. Anderson, "Visual and emotional salience influence eye movements," *Transactions on Applied Perception*, vol. 9, no. 3, pp. 13:1–13:18, 2012.

[306] R. M. Klein, "Inhibition of return," *Trends in Cognitive Sciences*, vol. 4, no. 4, pp. 138–147, 2000.

[307] J. Zhang and S. Sclaroff, "Saliency detection: A boolean map approach," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 153–160, December 2013.

[308] C. G. Gross, C. Bruce, R. Desimone, J. Fleming, and R. Gattass, "Cortical visual areas of the temporal lobe," in *Multiple Visual Areas* (C. N. Woolsey, ed.), vol. 2, pp. 187–216, New York, NY, USA: Humana Press, 1981.

[309] J. M. Henderson, "Eye movements and scene perception," in *The Oxford Handbook of Eye Movements* (S. P. Liversedge, I. D. Gilchrist, and S. Everling, eds.), ch. 12, pp. 593–606, Oxford, UK: Oxford University Press, 2011.

[310] R. Geirhos, D. H. J. Janssen, H. H. Schütt, J. Rauber, M. Bethge, and F. A. Wichmann, "Comparing deep neural networks against humans: object recognition when the signal gets weaker," *arXiv preprint 1706.06969*, 2017.

[311] M. Hutson, "Artificial intelligence faces reproducibility crisis," *Science*, vol. 359, no. 6377, pp. 725–726, 2018.

[312] A. Nuthmann, W. Einhäuser, and I. Schütz, "How well can saliency models predict fixation selecting in scenes beyond central bias? A new approach to model evaluation using generalized linear mixed models," *Frontiers in Human Neuroscience*, vol. 11, p. 491, 2017.

[313] E. Milios, M. Jenkin, and J. Tsotsos, "Design and performance of TRISH, a binocular robot head with torsional eye movements," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 7, no. 1, pp. 51–68, 1993.

[314] T. Parr and K. J. Friston, "The active construction of the visual world," *Neuropsychologia*, vol. 104, pp. 92 – 101, 2017.

[315] W. S. Geisler and J. S. Perry, "Real-time foveated multiresolution system for low-bandwidth video communication," in *Proc. of Photonics West Electronic Imaging*, pp. 294–305, International Society for Optics and Photonics, 1998.

[316] A. B. Watson, "A formula for human retinal ganglion cell receptive field density as a function of visual field location," *Journal of Vision*, vol. 14, no. 7, pp. 1–17, 2014.

Appendix A

SMILER YAML files

A.1 YAML files for Section 4.1

The following YAML file can be run with SMILER to reproduce the saliency maps used in Section 4.1.

Listing A.1: The YAML file to generate results oriented bar singletons.

```
1  experiment:
2    name: Orientation Singleton Search
3    description: Producing saliency maps for Section 4.1 of my Dissertation.
4    input_path: ../Arun320_bw/
5    base_output_path: ../Arun320_maps/
6    parameters:
7      center_prior: none
8
9  runs:
10   - algorithm: AIM
11
12   - algorithm: AWS
13
14   - algorithm: CAS
15
16   - algorithm: DGII
17
18   - algorithm: GBVS
19
20   - algorithm: ICF
21
22   - algorithm: IKN
23
24   - algorithm: IMSIG
25
26   - algorithm: oSALICON
27
```

```
28    - algorithm: SSR
29
30    - algorithm: SUN
```

A.2 YAML files for Section 4.2

The following YAML files can be run with SMILER to reproduce the saliency maps used in Section 4.2. Note that the person search arrays were combined within one folder and separated between canonical and flipped targets in subsequent processing, whereas the classical stimuli was separated by target and therefore required a specific YAML for each target-distractor pairing.

Listing A.2: The YAML file to generate results for a flipped A vs. canonical A's.

```
1  experiment:
2    name: Asymmetric Search
3    description: Producing saliency maps for Section 4.2 of my Dissertation.
4    input_path: ../AsymmetryImgs/A_flip/stimuli
5    base_output_path: ../AsymmetryMaps/A_flip
6    parameters:
7      center_prior: none
8
9  runs:
10   - algorithm: AIM
11
12   - algorithm: AWS
13
14   - algorithm: CAS
15
16   - algorithm: CVS
17
18   - algorithm: DGII
19
20   - algorithm: DVA
21
22   - algorithm: GBVS
23
24   - algorithm: ICF
25
26   - algorithm: IKN
27
28   - algorithm: IMSIG
29
30   - algorithm: QSS
31
32   - algorithm: RARE2012
33
34   - algorithm: SalGAN
35
```

```
36    - algorithm: oSALICON
37
38    - algorithm: SAM
39
40    - algorithm: SSR
```

Listing A.3: The YAML file to generate results for a canonical A vs. flipped A's.

```
1  experiment:
2    name: Asymmetric Search
3    description: Producing saliency maps for Section 4.2 of my Dissertation.
4    input_path: ../AsymmetryImgs/A_std/stimuli
5    base_output_path: ../AsymmetryMaps/A_std
6    parameters:
7      center_prior: none
8
9  runs:
10   - algorithm: AIM
11
12   - algorithm: AWS
13
14   - algorithm: CAS
15
16   - algorithm: CVS
17
18   - algorithm: DGII
19
20   - algorithm: DVA
21
22   - algorithm: GBVS
23
24   - algorithm: ICF
25
26   - algorithm: IKN
27
28   - algorithm: IMSIG
29
30   - algorithm: QSS
31
32   - algorithm: RARE2012
33
34   - algorithm: SalGAN
35
36   - algorithm: oSALICON
37
38   - algorithm: SAM
39
40   - algorithm: SSR
```

Listing A.4: The YAML file to generate results for a blue dot vs. magenta dots.

```
1  experiment:
```

```
  name: Asymmetric Search
  description: Producing saliency maps for Section 4.2 of my Dissertation.
  input_path: ../AsymmetryImgs/BvM/stimuli
  base_output_path: ../AsymmetryMaps/BvM
  parameters:
    center_prior: none

runs:
  - algorithm: AIM

  - algorithm: AWS

  - algorithm: CAS

  - algorithm: CVS

  - algorithm: DGII

  - algorithm: DVA

  - algorithm: GBVS

  - algorithm: ICF

  - algorithm: IKN

  - algorithm: IMSIG

  - algorithm: QSS

  - algorithm: RARE2012

  - algorithm: SalGAN

  - algorithm: oSALICON

  - algorithm: SAM

  - algorithm: SSR
```

Listing A.5: The YAML file to generate results for a magenta dot vs. blue dots

```
experiment:
  name: Asymmetric Search
  description: Producing saliency maps for Section 4.2 of my Dissertation.
  input_path: ../AsymmetryImgs/MvB/stimuli
  base_output_path: ../AsymmetryMaps/MvB
  parameters:
    center_prior: none

runs:
  - algorithm: AIM

```

```
12    - algorithm: AWS
13
14    - algorithm: CAS
15
16    - algorithm: CVS
17
18    - algorithm: DGII
19
20    - algorithm: DVA
21
22    - algorithm: GBVS
23
24    - algorithm: ICF
25
26    - algorithm: IKN
27
28    - algorithm: IMSIG
29
30    - algorithm: QSS
31
32    - algorithm: RARE2012
33
34    - algorithm: SalGAN
35
36    - algorithm: oSALICON
37
38    - algorithm: SAM
39
40    - algorithm: SSR
```

Listing A.6: The YAML file to generate results for a flipped Q vs. canonical Q's.

```
1  experiment:
2    name: Asymmetric Search
3    description: Producing saliency maps for Section 4.2 of my Dissertation.
4    input_path: ../AsymmetryImgs/Q_flip/stimuli
5    base_output_path: ../AsymmetryMaps/Q_flip
6    parameters:
7      center_prior: none
8
9  runs:
10   - algorithm: AIM
11
12   - algorithm: AWS
13
14   - algorithm: CAS
15
16   - algorithm: CVS
17
18   - algorithm: DGII
19
20   - algorithm: DVA
21
```

```
22    - algorithm: GBVS
23
24    - algorithm: ICF
25
26    - algorithm: IKN
27
28    - algorithm: IMSIG
29
30    - algorithm: QSS
31
32    - algorithm: RARE2012
33
34    - algorithm: SalGAN
35
36    - algorithm: oSALICON
37
38    - algorithm: SAM
39
40    - algorithm: SSR
```

Listing A.7: The YAML file to generate results for a canonical Q vs. flipped Q's.

```
1  experiment:
2    name: Asymmetric Search
3    description: Producing saliency maps for Section 4.2 of my Dissertation.
4    input_path: ../AsymmetryImgs/Q_std/stimuli
5    base_output_path: ../AsymmetryMaps/Q_std
6    parameters:
7      center_prior: none
8
9  runs:
10   - algorithm: AIM
11
12   - algorithm: AWS
13
14   - algorithm: CAS
15
16   - algorithm: CVS
17
18   - algorithm: DGII
19
20   - algorithm: DVA
21
22   - algorithm: GBVS
23
24   - algorithm: ICF
25
26   - algorithm: IKN
27
28   - algorithm: IMSIG
29
30   - algorithm: QSS
31
```

```
32    - algorithm: RARE2012
33
34    - algorithm: SalGAN
35
36    - algorithm: oSALICON
37
38    - algorithm: SAM
39
40    - algorithm: SSR
```

Listing A.8: The YAML file to generate results for an O vs. Q's.

```
1  experiment:
2    name: Asymmetric Search
3    description: Producing saliency maps for Section 4.2 of my Dissertation.
4    input_path: ../AsymmetryImgs/O_Q/stimuli
5    base_output_path: ../AsymmetryMaps/O_Q
6    parameters:
7      center_prior: none
8
9  runs:
10   - algorithm: AIM
11
12   - algorithm: AWS
13
14   - algorithm: CAS
15
16   - algorithm: CVS
17
18   - algorithm: DGII
19
20   - algorithm: DVA
21
22   - algorithm: GBVS
23
24   - algorithm: ICF
25
26   - algorithm: IKN
27
28   - algorithm: IMSIG
29
30   - algorithm: QSS
31
32   - algorithm: RARE2012
33
34   - algorithm: SalGAN
35
36   - algorithm: oSALICON
37
38   - algorithm: SAM
39
40   - algorithm: SSR
```

Listing A.9: The YAML file to generate results for a Q vs. O's.

```
experiment:
  name: Asymmetric Search
  description: Producing saliency maps for Section 4.2 of my Dissertation.
  input_path: ../AsymmetryImgs/Q_O/stimuli
  base_output_path: ../AsymmetryMaps/Q_O
  parameters:
    center_prior: none

runs:
  - algorithm: AIM

  - algorithm: AWS

  - algorithm: CAS

  - algorithm: CVS

  - algorithm: DGII

  - algorithm: DVA

  - algorithm: GBVS

  - algorithm: ICF

  - algorithm: IKN

  - algorithm: IMSIG

  - algorithm: QSS

  - algorithm: RARE2012

  - algorithm: SalGAN

  - algorithm: oSALICON

  - algorithm: SAM

  - algorithm: SSR
```

Listing A.10: The YAML file to generate results for person arrays.

```
experiment:
  name: Asymmetric Search
  description: Producing saliency maps for Section 4.2 of my Dissertation.
  input_path: ../AsymmetryImgs/coco_objects/stimuli
  base_output_path: ../AsymmetryMaps/coco_objects
  parameters:
    center_prior: none

```

```
runs:
  - algorithm: AIM

  - algorithm: AWS

  - algorithm: CAS

  - algorithm: CVS

  - algorithm: DGII

  - algorithm: DVA

  - algorithm: GBVS

  - algorithm: ICF

  - algorithm: IKN

  - algorithm: IMSIG

  - algorithm: QSS

  - algorithm: RARE2012

  - algorithm: SalGAN

  - algorithm: oSALICON

  - algorithm: SAM

  - algorithm: SSR
```

A.3 YAML files for Section 4.3

Listing A.11: The YAML file to generate results for the Action category of CAT2000.

```
experiment:
  name: CAT2000 Action
  description: Producing saliency maps for Section 4.3 of my Dissertation.
  input_path: /storage/Work/cat2k_stimuli/Action
  base_output_path: /storage/Work/cat2k_maps/Action

runs:
  - algorithm: AIM

  - algorithm: AWS

  - algorithm: CAS

```

```
14    - algorithm: CVS
15
16    - algorithm: DGII
17
18    - algorithm: DVA
19
20    - algorithm: GBVS
21
22    - algorithm: ICF
23
24    - algorithm: IKN
25
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.12: The YAML file to generate results for the Affective category of CAT2000.

```
1  experiment:
2    name: CAT2000 Affective
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Affective
5    base_output_path: /storage/Work/cat2k_maps/Affective
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
```

```
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.13: The YAML file to generate results for the Art category of CAT2000.

```
1  experiment:
2    name: CAT2000 Art
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Art
5    base_output_path: /storage/Work/cat2k_maps/Art
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
```

```
38    - algorithm: SSR
```

Listing A.14: The YAML file to generate results for the BlackWhite category of CAT2000.

```
1  experiment:
2    name: CAT2000 BlackWhite
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/BlackWhite
5    base_output_path: /storage/Work/cat2k_maps/BlackWhite
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
38   - algorithm: SSR
```

Listing A.15: The YAML file to generate results for the Cartoon category of CAT2000.

```
1  experiment:
2    name: CAT2000 Cartoon
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Cartoon
5    base_output_path: /storage/Work/cat2k_maps/Cartoon
6
7  runs:
```

```
8     - algorithm: AIM
9
10    - algorithm: AWS
11
12    - algorithm: CAS
13
14    - algorithm: CVS
15
16    - algorithm: DGII
17
18    - algorithm: DVA
19
20    - algorithm: GBVS
21
22    - algorithm: ICF
23
24    - algorithm: IKN
25
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.16: The YAML file to generate results for the Fractal category of CAT2000.

```
1  experiment:
2    name: CAT2000 Fractal
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Fractal
5    base_output_path: /storage/Work/cat2k_maps/Fractal
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
```

```
20    - algorithm: GBVS
21
22    - algorithm: ICF
23
24    - algorithm: IKN
25
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.17: The YAML file to generate results for the Indoor category of CAT2000.

```
1  experiment:
2    name: CAT2000 Indoor
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Indoor
5    base_output_path: /storage/Work/cat2k_maps/Indoor
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
```

```
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.18: The YAML file to generate results for the Inverted category of CAT2000.

```
1  experiment:
2    name: CAT2000 Inverted
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Inverted
5    base_output_path: /storage/Work/cat2k_maps/Inverted
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
38   - algorithm: SSR
```

Listing A.19: The YAML file to generate results for the Jumbled category of CAT2000.

```
1  experiment:
```

```
2    name: CAT2000 Jumbled
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Jumbled
5    base_output_path: /storage/Work/cat2k_maps/Jumbled
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
38   - algorithm: SSR
```

Listing A.20: The YAML file to generate results for the LineDrawing category of CAT2000.

```
1  experiment:
2    name: CAT2000 LineDrawing
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/LineDrawing
5    base_output_path: /storage/Work/cat2k_maps/LineDrawing
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
```

```
14    - algorithm: CVS
15
16    - algorithm: DGII
17
18    - algorithm: DVA
19
20    - algorithm: GBVS
21
22    - algorithm: ICF
23
24    - algorithm: IKN
25
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.21: The YAML file to generate results for the LowResolution category of CAT2000.

```
1  experiment:
2    name: CAT2000 LowResolution
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/LowResolution
5    base_output_path: /storage/Work/cat2k_maps/LowResolution
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
```

```
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.22: The YAML file to generate results for the Noisy category of CAT2000.

```
1  experiment:
2    name: CAT2000 Noisy
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Noisy
5    base_output_path: /storage/Work/cat2k_maps/Noisy
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
```

```
38    - algorithm: SSR
```

Listing A.23: The YAML file to generate results for the Object category of CAT2000.

```
1  experiment:
2    name: CAT2000 Object
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Object
5    base_output_path: /storage/Work/cat2k_maps/Object
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
38   - algorithm: SSR
```

Listing A.24: The YAML file to generate results for the OutdoorManMade category of CAT2000.

```
1  experiment:
2    name: CAT2000 OutdoorManMade
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/OutdoorManMade
5    base_output_path: /storage/Work/cat2k_maps/OutdoorManMade
6
7  runs:
```

```
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
38   - algorithm: SSR
```

Listing A.25: The YAML file to generate results for the OutdoorNatural category of CAT2000.

```
1  experiment:
2    name: CAT2000 OutdoorNatural
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/OutdoorNatural
5    base_output_path: /storage/Work/cat2k_maps/OutdoorNatural
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
```

```
20    - algorithm: GBVS
21
22    - algorithm: ICF
23
24    - algorithm: IKN
25
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.26: The YAML file to generate results for the Pattern category of CAT2000.

```
1  experiment:
2    name: CAT2000 Pattern
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Pattern
5    base_output_path: /storage/Work/cat2k_maps/Pattern
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
```

```
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.27: The YAML file to generate results for the Random category of CAT2000.

```
1  experiment:
2    name: CAT2000 Random
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Random
5    base_output_path: /storage/Work/cat2k_maps/Random
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
38   - algorithm: SSR
```

Listing A.28: The YAML file to generate results for the Satelite category of CAT2000.

```
1  experiment:
```

```
2    name: CAT2000 Satelite
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Satelite
5    base_output_path: /storage/Work/cat2k_maps/Satelite
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
26   - algorithm: IMSIG
27
28   - algorithm: QSS
29
30   - algorithm: RARE2012
31
32   - algorithm: SalGAN
33
34   - algorithm: oSALICON
35
36   - algorithm: SAM
37
38   - algorithm: SSR
```

Listing A.29: The YAML file to generate results for the Sketch category of CAT2000.

```
1  experiment:
2    name: CAT2000 Sketch
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Sketch
5    base_output_path: /storage/Work/cat2k_maps/Sketch
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
```

```
14    - algorithm: CVS
15
16    - algorithm: DGII
17
18    - algorithm: DVA
19
20    - algorithm: GBVS
21
22    - algorithm: ICF
23
24    - algorithm: IKN
25
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Listing A.30: The YAML file to generate results for the Social category of CAT2000.

```
1  experiment:
2    name: CAT2000 Social
3    description: Producing saliency maps for Section 4.3 of my Dissertation.
4    input_path: /storage/Work/cat2k_stimuli/Social
5    base_output_path: /storage/Work/cat2k_maps/Social
6
7  runs:
8    - algorithm: AIM
9
10   - algorithm: AWS
11
12   - algorithm: CAS
13
14   - algorithm: CVS
15
16   - algorithm: DGII
17
18   - algorithm: DVA
19
20   - algorithm: GBVS
21
22   - algorithm: ICF
23
24   - algorithm: IKN
25
```

```
26    - algorithm: IMSIG
27
28    - algorithm: QSS
29
30    - algorithm: RARE2012
31
32    - algorithm: SalGAN
33
34    - algorithm: oSALICON
35
36    - algorithm: SAM
37
38    - algorithm: SSR
```

Appendix B

Additional STAR-FC Details and Experiments

B.1 Retinal Transform

The retinal transform model used by STAR-FC has been modified slightly during the development of the model. The version used to produce the results shown in Section 5.5.1 is described in detail in [97]. The version used for all subsequent results includes a few minor changes and is described below. The largest change is that the generalized Gamma distribution was refitted to better simulate rod vision, and the interpolation function was reimplemented in CUDA to improve the speed of execution.

As with most retinal transform models, our approach starts by building a Gaussian pyramid and then for each pixel the appropriate level of the pyramid is sampled depending on how far the pixel is from the current gaze point. The level of the pyramid to sample from is computed as follows

$$L_{x,y} = \frac{\frac{\pi}{180}(\text{atan}((D_{\text{rad}} + \text{dotpitch})\frac{1}{D_{\text{view}}}) - \text{atan}((D_{\text{rad}} - \text{dotpitch})\frac{1}{D_{\text{view}}}))}{(\epsilon_2(\alpha * (EC + \epsilon_2))). * log(\frac{1}{CT_0})} \tag{B.1}$$

where D_{rad} is the radial distance between the point (x, y) and the current gaze point, EC is the eccentricity in degrees from the fovea centre for point (x, y), dotpitch is the size of the pixel of the monitor in meters and D_{view} refers to the viewing distance. α, ϵ_2 and CT_0 are constants from [315]. In this equation the numerator represents the maximum spatial frequency that can be represented

at the given distance from the current gaze point and the denominator is the maximum spatial resolution that can be resolved by the eye.

However, [315] only considered cone vision. For more biologically accurate results we augment this model with rod vision using the generalized Gamma distribution as proposed by Watson [316]. Since Watson used cell counts to fit a distribution function, we adjust the parameters to convert the model to the units that we use, namely the levels of pyramid. Therefore, we set the parameters of the generalized Gamma distribution as follows: $\alpha = 2.46, \beta = 121.8, \gamma = 0.77, \sigma = 861.27$ and $\mu = -1$. To find corresponding levels of the pyramid for each pixel we compute the generalized gamma distribution for EC and plug it into the equation (1) as the denominator.

Finally we compute the retinally transformed image using a bi-cubic interpolation routine for 3D volumes to sample the required level of the pyramid for each pixel using a CUDA reimplementation of the `ba_interp3` function [1].

Note that the image is first converted to YCrCb colour space, and we compute cone distribution for each colour channel separately but rod distribution only for the intensity channel, since rods do not contribute to colour vision. Rod and cone functions contribute 40% and 60%, respectively, to the final transformed image.

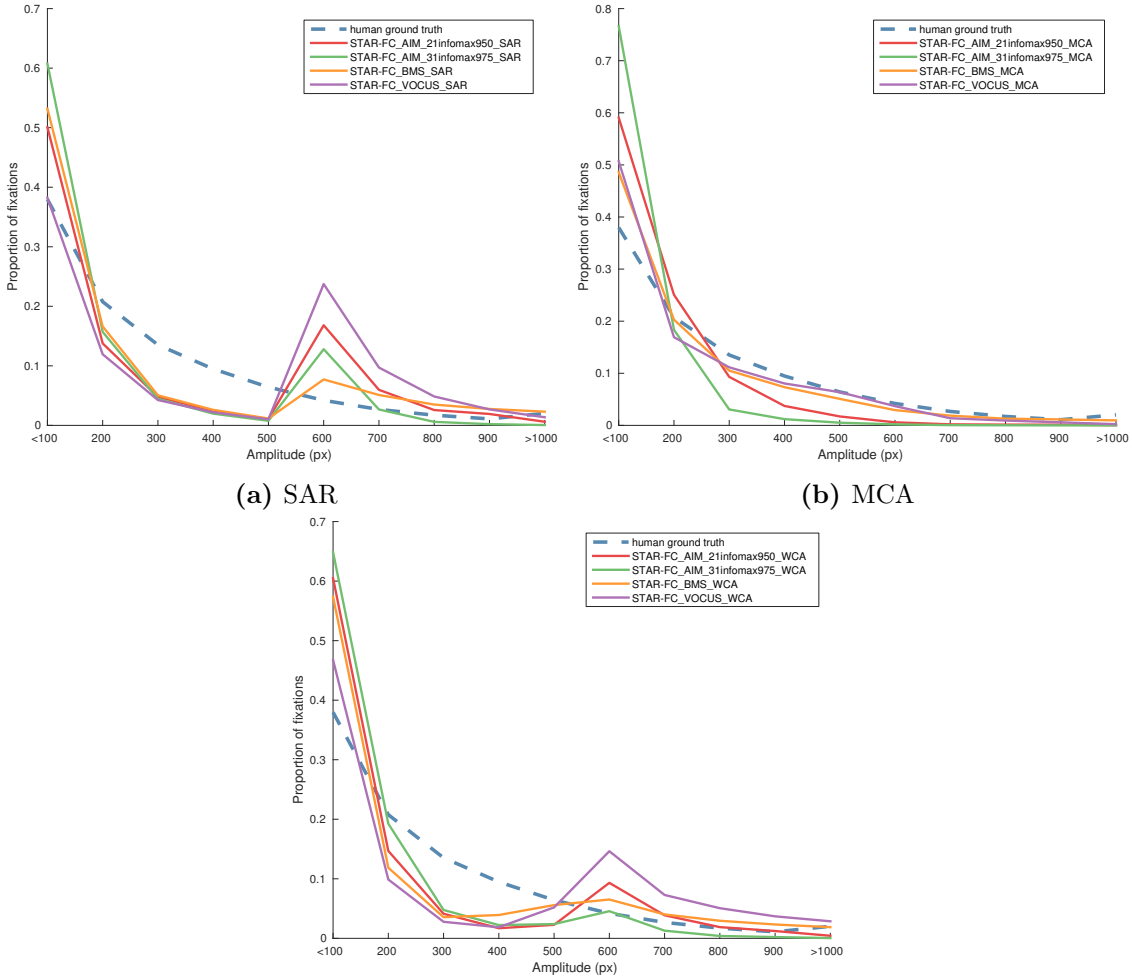
The viewing conditions in all our experiments match the experimental conditions reported for the CAT2000 dataset [12], namely all stimuli span 45 degrees and the viewing distance is set to 1.06 m.

B.2 Saccade amplitudes

Figure B.1 shows plots of the fixation amplitudes that demonstrate the effect of using different saliency algorithms in the periphery and blending strategies. Both SAR and WCA blending strategies (Figure B.1a and Figure B.1c) lead to a pronounced spike in the distribution, which approximately corresponds to the diameter of the central field. MCA, on the other hand, produces a more even distribution of the amplitudes (Figure B.1b). Figure B.2 shows fixation amplitudes for all tested bottom-up saliency algorithms. Note that all of them greatly underestimate the number of short saccades (less than 100 px) and generally have a much flatter fall off than the human ground

[1] `https://www.mathworks.com/matlabcentral/fileexchange/21702-3d-volume-interpolation-with-ba-interp3--fast-interp3-replacement`

truth distribution.



(a) SAR **(b)** MCA

(c) WCA

Figure B.1: Plots of fixation amplitudes demonstrating the effects of different strategies for combining peripheral and central fields of STAR-FC (SAR, MCA and WCA), and different bottom-up saliency algorithms in the peripheral field (AIM, VOCUS and BMS).

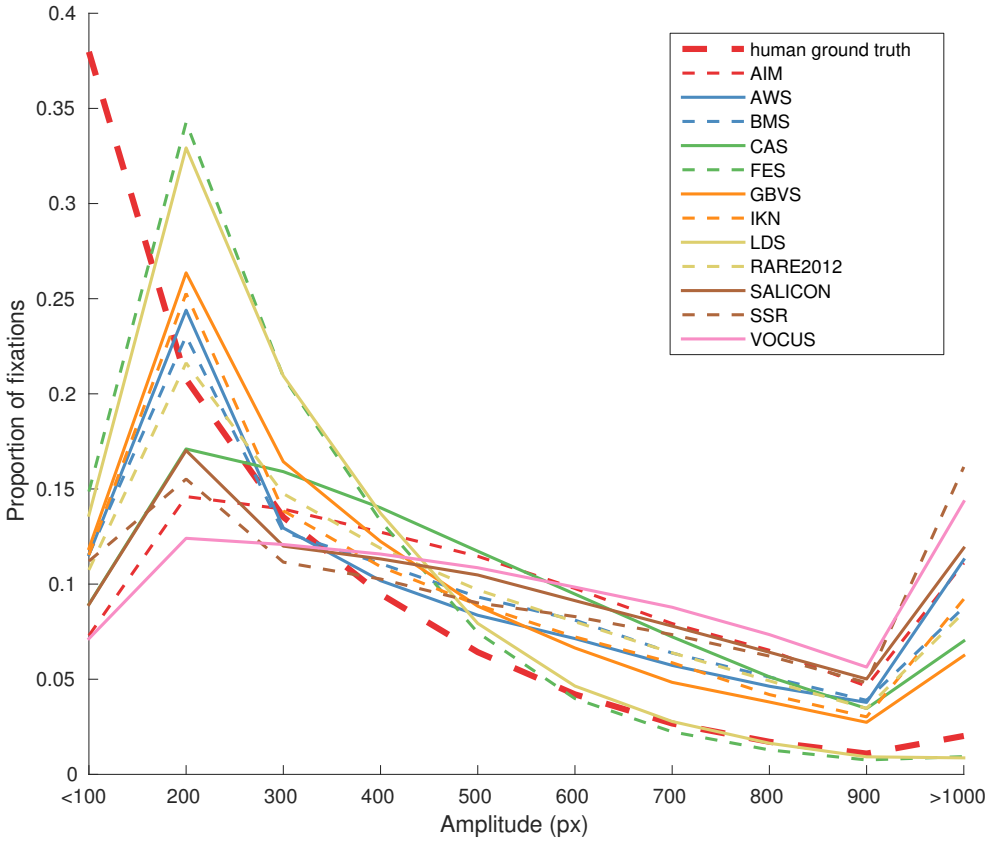


Figure B.2: Fixation amplitudes for 12 state-of-the-art bottom-up saliency algorithms.

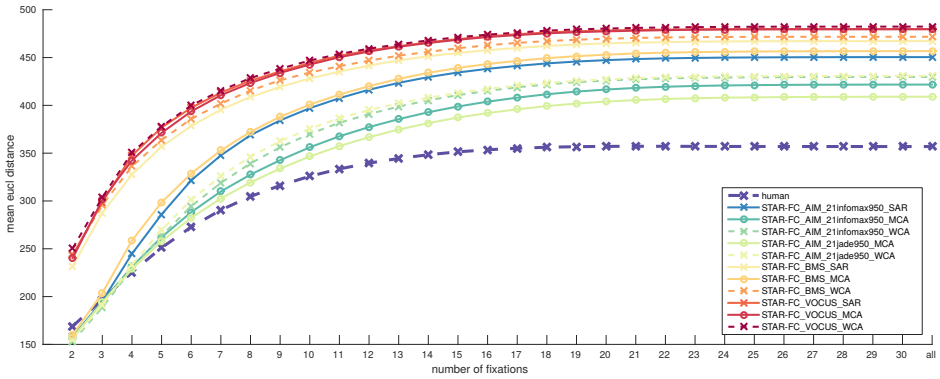
B.3 Trajectory Scores

Figure B.3 and Figure B.4 show trajectory scores for full sequence length for various STAR-FC variants and all tested saliencey algorithms. Note that as the sequences get longer they begin to diverge and the trajectory scores saturate.

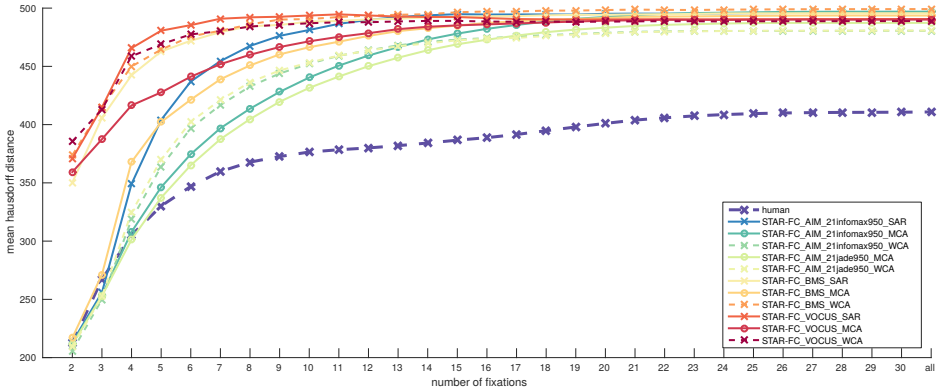
Figure B.5 shows the AUC score for the first 5 fixations for all saliency algorithms and the best performing STAR-FC formulation (using AIM with 21infomax950 basis and MCA blending strategy) split by category. For human fixations we report the AUC for the average pairwise distance. As noted in Chapter 5, it correlates well with inter-observer consistency for different categories of images (e.g. high inter-observer consistency for Sketch translates to a smaller average pairwise distances across all metrics, whereas the opposite is true for the Jumbled category).

Note that saliency algorithms tend to follow the same trends as human inter-observer scores. In general, categories with high inter-observer consistency such as Affective or Sketch are not as challenging as categories with lower human to human fixation consistency. One major exception

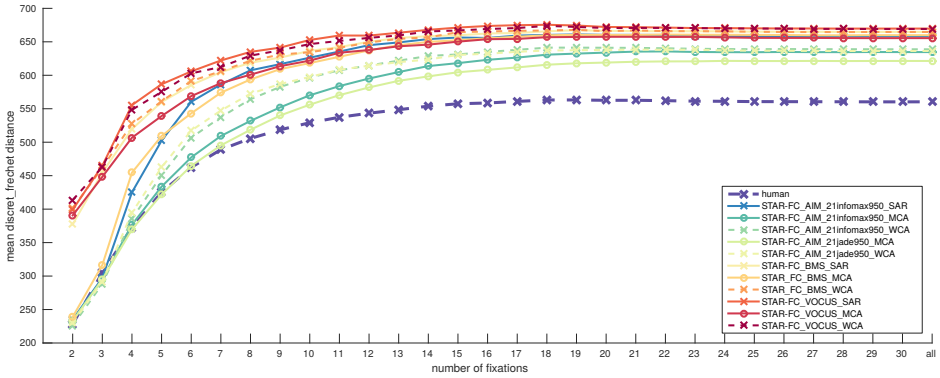
is the Low Resolution category, which has a high degree of inter-observer consistency but is nevertheless extremely challenging for all saliency algorithms. This calls for more investigation of the effect that blurring has on the quality of saliency prediction.



(a) Euclidean distance

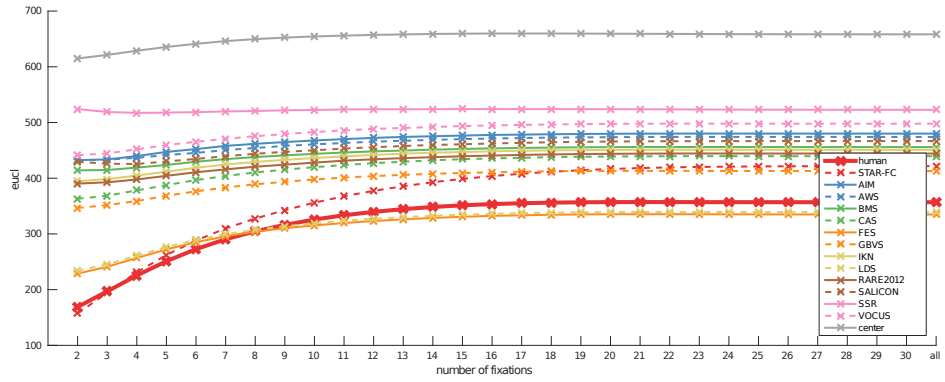


(b) Hausdorff distance

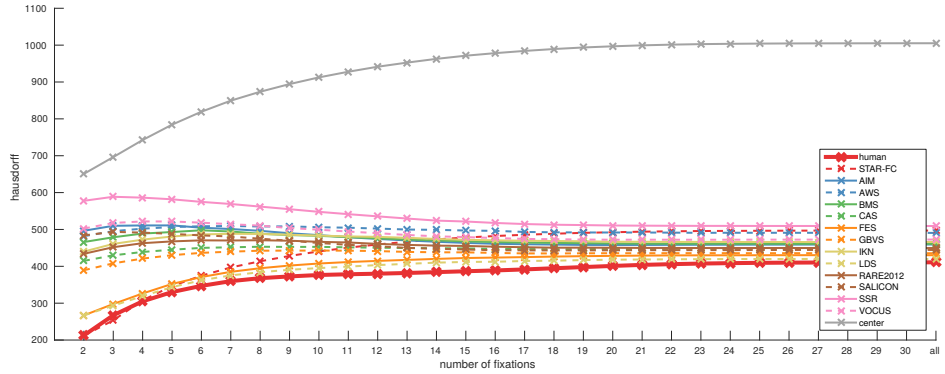


(c) Frechet distance

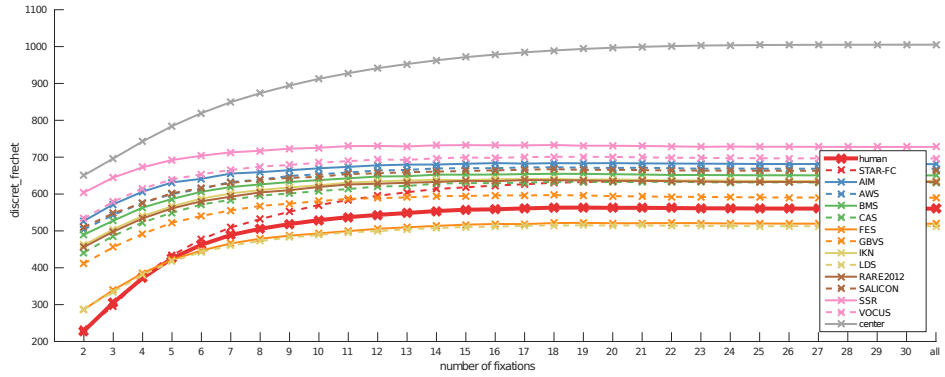
Figure B.3: A comparison of fixation prediction scores over the full length of the fixation sequences for variants of STAR-FC.



(a) Euclidean distance

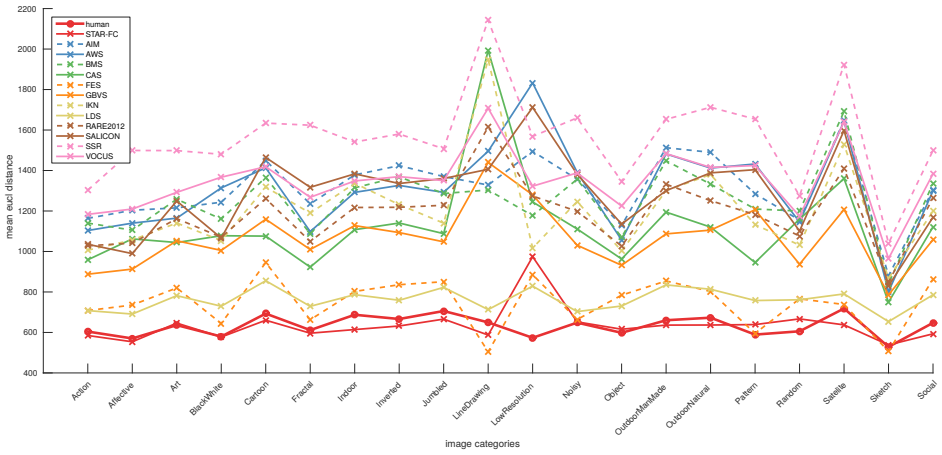


(b) Hausdorff distance

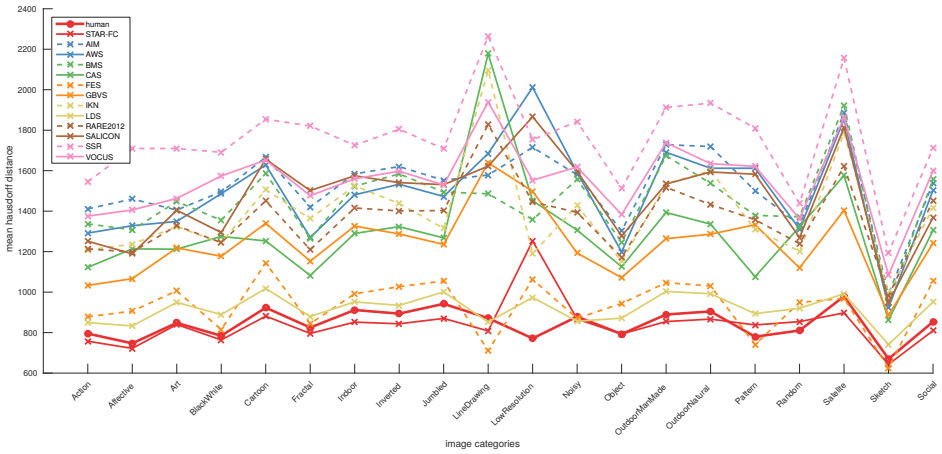


(c) Frechet distance

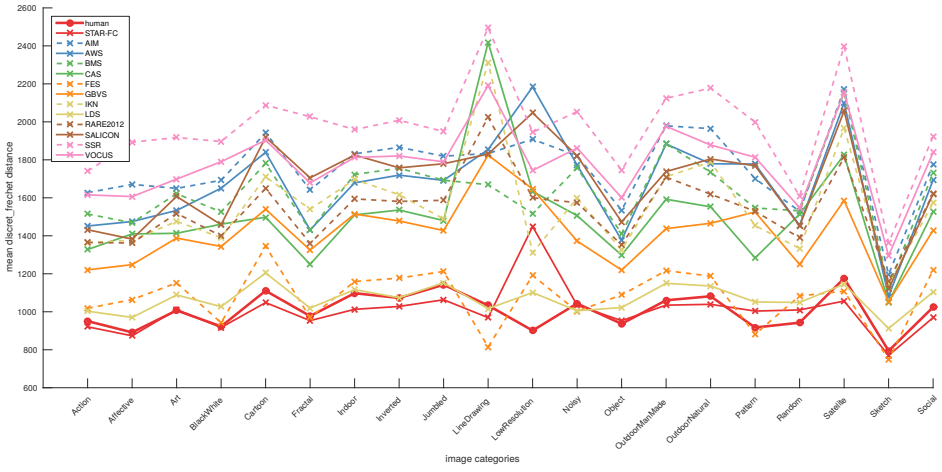
Figure B.4: A comparison of fixation prediction scores over the full length of the fixation sequences for all tested saliency algorithms and the best performing STAR-FC model (using AIM with 21infomax950 basis in the peripheral field and MCA blending strategy).



(a) Euclidean distance



(b) Hausdorff distance



(c) Frechet distance

Figure B.5: Fixation prediction scores for all tested saliency algorithms and the best performing STAR-FC model (using AIM with 21infomax950 basis in the peripheral field and MCA blending strategy). For each category we measured the mean distance from the human fixation and plotted the area-under-the-curve (AUC) score for the first 5 fixations.

B.4 2D Histograms of Fixations

Figure B.6 shows 2D histograms of fixations for all saliency algorithms with MSE scores.

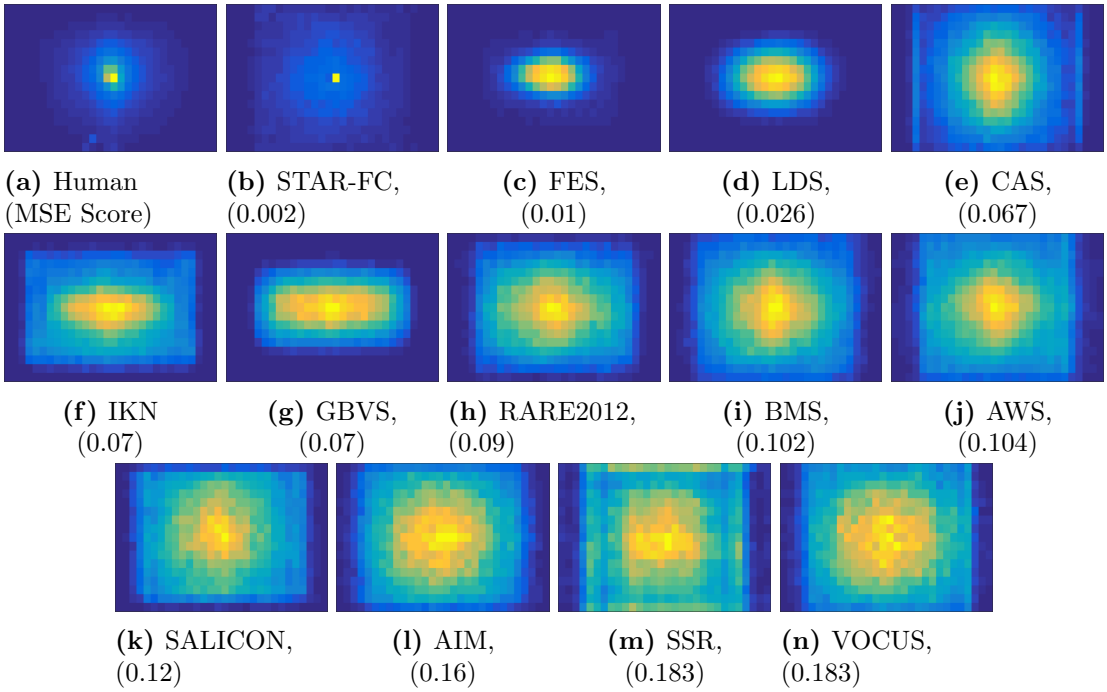


Figure B.6: 2D histograms of fixation locations over the CAT2000 dataset for all tested bottom-up saliency algorithms. Mean-squared-error (MSE) scores between model and human distributions are shown in parentheses under each model name. The algorithms are sorted by MSE in ascending order, starting with STAR-FC (AIM with 21infomax950 basis and MCA blending strategy), which is an order of magnitude closer to the human distribution than the best bottom-up algorithm (FES).

B.5 Examples of Predicted Fixation Sequences

Below are some selected examples of predicted fixations. For clarity we only compare STAR-FC and one saliency algorithm at a time and show only the closest human sequences to each of the predicted sequences. Furthermore, we show results only for the first 3 and 5 fixations. We selected FES as the top performing contrast-based algorithm and SALICON as the top performing CNN-based algorithm for comparison with the best performing STAR-FC formulation (21infomax950 bases and MCA blending strategy).

In Figure B.7 examples from categories with high inter-observer consistency (Affective and Low Resolution) are shown. Figure B.8 shows examples from the "Satelite" category which has low

inter-observer consistency.

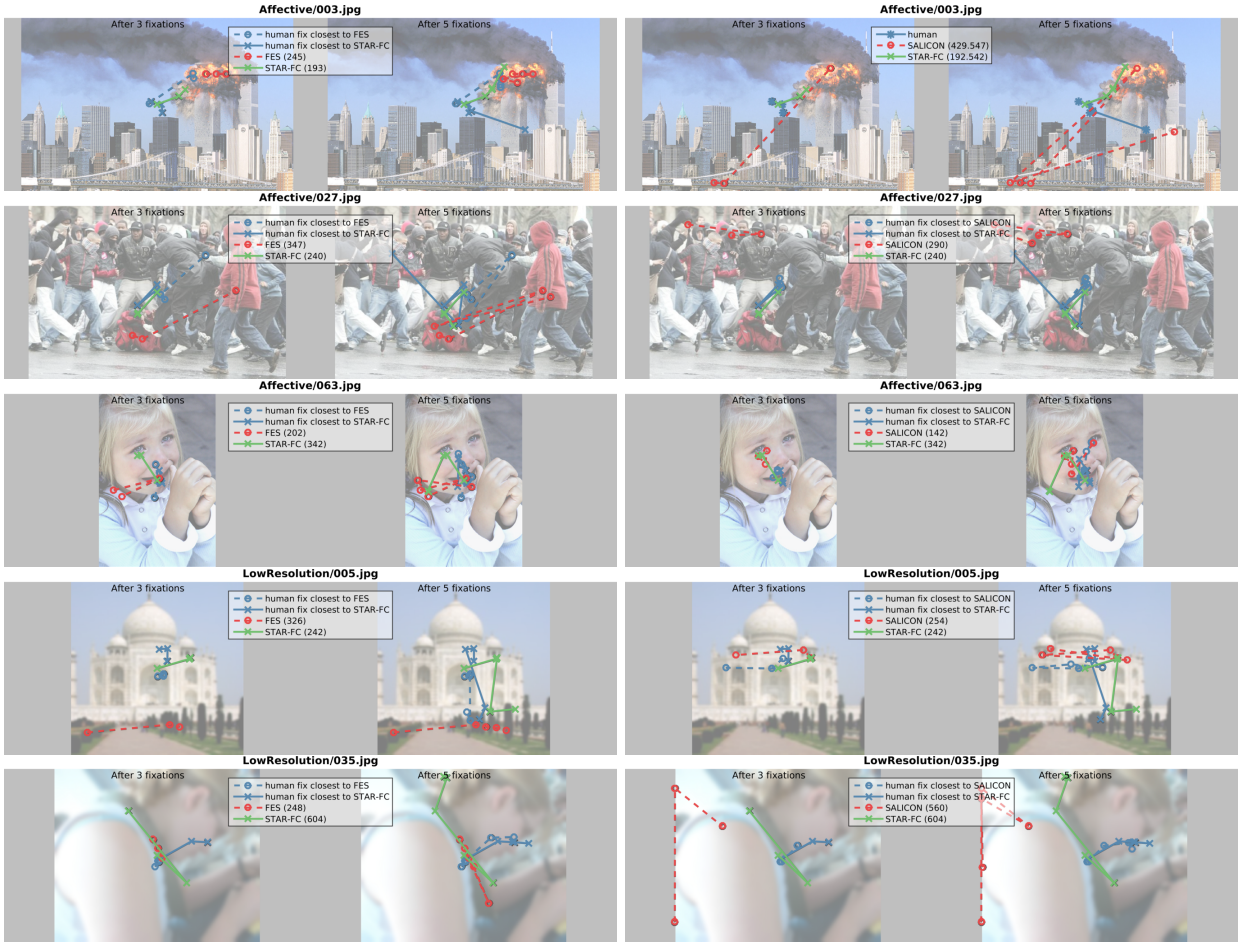


Figure B.7: Examples of fixations predicted by FES (left column) and SALICON (right column) compared to the proposed STAR-FC model (with AIM 21infomax basis and MCA blending strategy). Blue lines represent the closest human fixations to each of the compared algorithms and numbers in parentheses indicate the corresponding Euclidean distance.

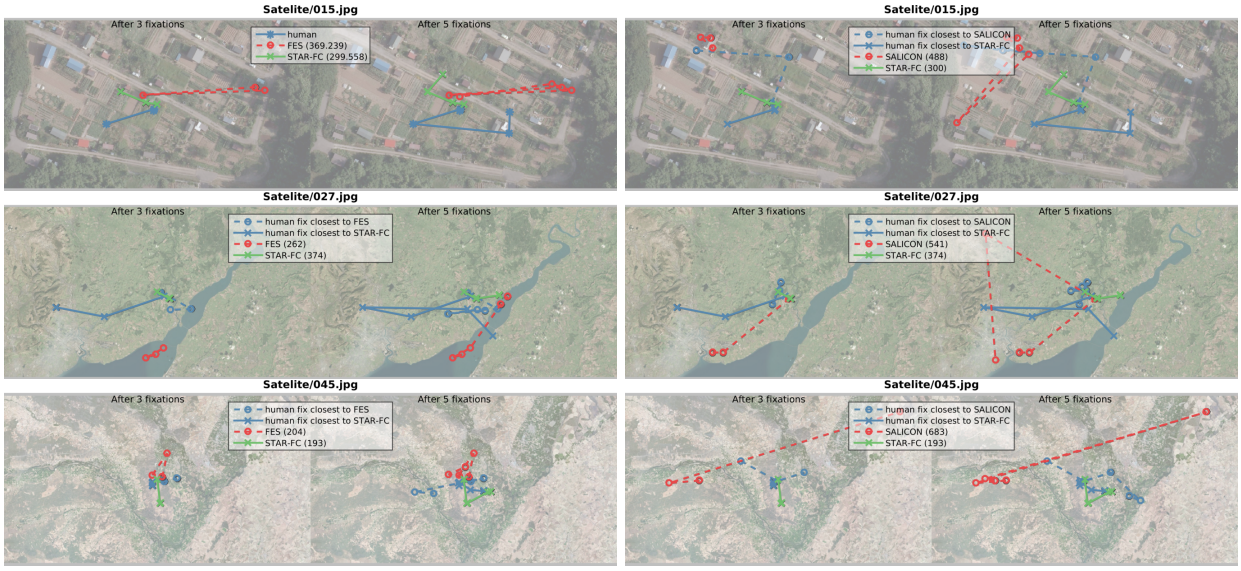


Figure B.8: Examples of fixations predicted by FES (left column) and SALICON (right column) compared to the proposed STAR-FC model (with AIM 21infomax basis and MCA blending strategy). Blue lines represent the closest human fixations to each of the compared algorithms and numbers in parentheses indicate the corresponding Euclidean distance.