Low-Cost Eye Gaze Prediction System for Interactive Networked Video Streaming

Yunlong Feng *Student Member, IEEE*, Gene Cheung *Senior Member, IEEE*,
Wai-tian Tan *Senior Member, IEEE*, Patrick Le Callet *Senior Member, IEEE*,
Yusheng Ji *Member, IEEE*

Abstract

Eye gaze is now used as a content adaptation trigger in interactive media applications, such as customized advertisement in video, and bit allocation in streaming video based on region-of-interest (ROI). The reaction time of a gaze-based networked system, however, is lower-bounded by the network round trip time (RTT). Furthermore, only low-sampling-rate gaze data is available when commonly available webcam is employed for gaze tracking. To realize responsive adaptation of media content even under non-negligible RTT and using common low-cost webcams, we propose a Hidden Markov Model (HMM) based gaze-prediction system that utilizes the visual saliency of the content being viewed. Specifically, our HMM has two states corresponding to two of human's intrinsic gaze behavioral movements, and its model parameters are derived offline via analysis of each video's visual saliency maps. Due to the strong prior of likely gaze locations offered by saliency information, accurate runtime gaze prediction is possible even under large RTT and using common webcam.

We demonstrate the applicability of our low-cost gaze prediction system by focusing on ROI-based bit allocation for networked video streaming. To reduce transmission rate of a video stream without degrading viewer's perceived visual quality, we allocate more bits to encode the viewer's current spatial ROI, while devoting fewer bits in other spatial regions. The challenge lies in overcoming the delay between the time a viewer's ROI is detected by gaze tracking, to the time the effected video is encoded, delivered and displayed at the viewer's terminal. To this end, we use our proposed low-cost gaze prediction system to predict future eye gaze locations, so that optimized bit allocation can be performed for future frames. Through extensive subjective testing, we show that bit-rate can be reduced by up to 29% without noticeable visual quality degradation when RTT is as high as 200ms.

Manuscript received April 8, 2013; created Jul 26, 2012, resubmitted Oct 11, 2012, revised Dec 10, 2012.

Yunlong Feng is with the Graduate University for Advanced Studies (SOKENDAI), and National Institute of Informatics, Tokyo Japan. E-mail: fengyl@nii.ac.jp

Gene Cheung and Yusheng Ji are with National Institute of Informatics, 2-1-2, Hitotsubashi, Chiyoda-ku, Tokyo Japan, 101-8430. E-mail: {cheung, kei}@nii.ac.jp

Wai-tian Tan is with Hewlett-Packard Laboratories, 1501 Page Mill Road, M/S 1181, Palo Alto, CA 94304. Phone: +1-650-857-3844. Fax: +1-650-857-8491. E-mail: wai-tian.tan@hp.com

Patrick Le Callet is with Polytech Nantes/Universite de Nantes, IRCCyN/IVC. rue Christian Pauc La Chantretir, BP 50609 44306 Nantes Cedex 3. Phone: +33 (0)2 40 68 30 47. E-mail: patrick.lecallet@univ-nantes.fr

I. Introduction

Eye gaze tracking—the inference of a viewer's point of visual focus based on camera-captured images of the eye(s)—has been intensively studied in the last decade [1, 2], to the level of maturity that it is now a commercially available technology [3, 4]. To unlock the potential of this new tool, many applications now employ eye gaze as a content adaptation trigger for media interaction. One example is large display customization [5], where the visual content rendered is adaptively composed (e.g., insert customized advertisements) according to tracked past and current gaze locations. Another example is immersive gaming [6], where different animated non-player characters (NPC) react differently depending on which NPC the viewer is currently looking at and showing facial expressions.

For networked media systems, gaze data are collected at a client in real-time and sent to a server to effect changes in media content. The reaction time of the gazed-based trigger, however, is lower-bounded by the round trip time (RTT) of the transmission networks. For today's Internet, RTT can be as large as 200ms, which significantly exceeds the 60ms threshold [7] for tolerable lag between a change in viewer's visual focus and the corresponding content update in *gaze-contingent displays* (GCD) [8]. This large RTT delay severely limits the efficacy of gaze-based networked media systems. Hence, predictive strategies are necessary for effective application of eye gaze to networked interactive media systems.

In this paper, we propose a low-cost gaze prediction system using our proposed Hidden Markov Model (HMM) to predict viewer's gaze location in the future (RTT seconds from the present), so that the server can adapt media content using the predicted gaze locations instead of the most recently tracked gaze locations, reducing end-to-end reaction delay. The key idea is to *establish correlation between tracked eye-gaze movements and the current video content being watched*, so that future gaze locations can be predicted with the help of content analysis of video that is about to be displayed. Such analysis can be performed offline computation-efficiently. Specifically, we first design an HMM with two latent states that correspond to two of human's intrinsic gaze behavioral movements: *tracking* and *saccade* [9]. Tracking means a viewer is following the movement of an identifiable object in video. Saccade means a viewer is shifting his visual attention from one object of interest to another. Thus, if a viewer following an object in tracking state, then his future gaze location will likely be correlated with the future position of the object.

HMM parameters (most importantly, state transition probabilities) are derived offline at server on a per-video basis via analysis of the video's *visual saliency maps* [10–12]. In bottom-up visual saliency models, by computing weighted combinations of detected low-level features in a video frame such as lighting / color contrast, flicker, motion, etc, a saliency map reveals, as a first order approximation, the amount of visual attention (saliency) each spatial region in the frame will draw from the viewer. By analyzing how spatial saliency in video frames changes over time, we can estimate the regions-of-interest (ROI) a viewer may choose to observe and how he may switch ROIs over time, resulting in HMM state transition probabilities. Through saliency map analysis, we can also partition the video into temporal segments of roughly stationary gaze statistics—each a set of consecutive frames that induce observer's gaze movements well described statistically by the same set of HMM parameters. During actual streaming, a window of noisy gaze observations are collected in real-time for a forward algorithm (FA) to compute the most likely current latent state. Given the deduced HMM state, gaze prediction using Kalman filtering [13] is performed to predict gaze location RTT into the future to reactively effect media content adaptation at server.

We demonstrate the applicability of our gaze prediction strategy through a networked video streaming application that performs bit allocation based on ROI. In face of limited network transmission bandwidth, the conventional end-to-end streaming approach [14, 15] is to throttle sending rate, so that limited network bandwidth can be properly shared among competing users. Reduction of sending rate, however, causes a proportional degradation in video quality due to more aggressive signal quantization, often resulting in unacceptable visual experience.

One can alleviate this bandwidth-constrained problem by exploiting unique characteristics of the human perceptual system [7, 8, 16]. In particular, it has been shown [16, 17] that viewer's ability to perceive details away from the current gaze focal point falls precipitously as the angle away from the focal point increases. Thus, a smart bit allocation scheme [18, 19] can allocate more bits to ROI to minimize noticeable quantization noise, and fewer bits elsewhere. In this way, the *perceived* video quality remains the same while encoded bit-rate can be decreased. The technical challenge, however, is to overcome the unavoidable delay from the time a ROI is estimated, to the time the corresponding effected change in video bit allocation is executed, transmitted and rendered on the viewer's terminal. To overcome RTT delay, we use our proposed gaze prediction system to predict future gaze locations, so that optimal bit allocation can be performed for future frames. Experiments using our developed real-time video coding and streaming system, integrated with an off-the-shelf web camera and a software gaze tracker [20], show that transmission rate can be reduced by up to 29% without loss of perceived video quality for RTT as high as 200ms.

The outline of the paper is as follows. We first discuss related work in Section II. We then discuss our proposed HMM for eye-gaze prediction in Section III. We discuss how HMM parameters are derived via analysis of visual saliency maps in Section IV. For a given estimated HMM state, we discuss how we predict gaze location RTT into the future in Section V. Having obtained a gaze prediction, the corresponding bit allocation scheme is discussed in Section VI. Experimentation and conclusions are discussed in Section VII and Section VIII, respectively.

II. Related Work

We divide our discussion on related works into three sections: i) previous work in eye-gaze prediction, ii) previous works in visual saliency maps, and iii) previous work in ROI-based bit allocation for video coding / streaming. Finally, we discuss the novelty of this paper relative to our previous work on gaze prediction.

A. *Eye-gaze Prediction*

While eye-gaze tracking has been studied extensively in the literature [1, 2]—including newer systems that do not require active calibration [21, 22]—there are relatively few prior work on eye-gaze prediction. Assuming a viewer's eye-gaze movements are either fixation or saccade, [23] first proposed a Kalman-filter-based eye-gaze movement prediction scheme to predict viewer's gaze location in the future. The same authors later improved their model by integrating it with a linear horizontal oculomotor plant mechanical model, a detailed motion model to predict eye movements based on the mechanics of the human eye using a large number of parameters [24, 25].

Our gaze-prediction strategy differs from [24, 25] in two major respects. First, rather than modeling the mechanics of the human eye, we approach the gaze prediction problem from a pure statistical learning perspective, where our two-state HMM is simple and maps intuitively to two of human's intrinsic gaze behavioral movements. Second, unlike [24, 25] which predicted gaze movements in a content-independent manner, the major insight in our approach is to establish correlation between eye-gaze movements and the video content being watched. We do so because it has been shown in numerous subjective experiments in a variety of viewing scenarios [10, 26] that human visual attention is very often driven by innate visual stimulus in the observed content. Hence it is quite reasonable to assume that the aforementioned correlation exists and can be exploited for gaze prediction. This content-dependent approach has two implications: i) we only need to estimate very few parameters in a simple HMM model, and ii) only coarsely sampled gaze data are required to estimate the HMM state (tracking or saccade) an observer's gaze is currently in, so that a low-cost web camera capturing video at low frame rate (30 fps was used in our system) can be used in place of more expensive standalone gaze trackers used in [24, 25], lowering the barrier to mass deployment[1].

B. Visual Saliency Map

Visual attention (VA) modeling has focused many research efforts in the last decade following up efforts from the community of vision science and perception to better understand the fundamentals of visual attention. Several computational models to emulate VA have been consequently proposed, detecting the locations that attract the eye gaze. Most of the models compute a saliency map that values each pixel according to its visual saliency. While top-down visual saliency modeling is also possible [27], we focus our discussion in bottom-up visual attention process.

Several approaches, more or less biological, have been proposed. All the approaches share the same main principle: saliency is closely related to singularity or rareness. They can be classified into three different categories:

1) Hierarchical models [10, 12, 28, 29] based on computational architecture characterized by a hierarchical decomposition followed by ad hoc processing on each sub-band (e.g. DOG to mimic receptor field properties to seek for singularities) to estimate the salience. Different techniques are then used to aggregate this information across levels in order to build a unique saliency map.
2) Statistical models [30–32] based on probabilistic analysis of the content. Following the plausible link between saliency and singularity, the saliency at a given location is defined as a measure of the deviation between features at this location with respect to its neighborhood.

[1] We note that because our low-cost gaze prediction system only makes predictions when the estimated state is tracking (saccade is deemed too complex to predict given the low sampling rate), the intended interactive media applications are limited to non-mission-critical ones, such as ROI bit allocation as detailed in this paper, and others as described in the first paragraph in the Introduction.

3) Bayesian models [33, 34] are useful to introduce prior-knowledge (e.g. contextual information like statistic of natural scene) and another alternative to cope with the saliency/singularity link. For instance, Itti and Baldi [34] introduced a Bayesian definition of surprise in order to measure the distance between posterior and prior beliefs of the observers. They proved that this measure, the surprise, is related to visual attention.

The quantitative assessment of the performances of these different models is still an open issue, but it appears that all these models reach similar results, whatever the assessment technique [35]. Our goal here is not to propose new visual saliency maps, but to use saliency maps, computed using previously established techniques, to derive HMM parameters offline in a computationally efficient way. This motivation is not unlike previous proposals that use saliency maps to resolve uncertainty in gaze estimates [21, 22, 36], except that our derived HMM parameters reflect the *temporal* aspect of expected gaze behavior, rather than the *spatial* aspect. In this paper, we selected methodology in [10] to compute saliency maps, based on a plausible model of bottom-up visual attention. Considering previous comments on performance, this model offers good performance with reasonable computational cost. An existing implementation of the model is available at [37]. We note, however, that our proposed gaze prediction strategy is agnostic to the particular type of saliency model, and thus can be made interoperable to other saliency models such as [27].

C. ROI-based Bit Allocation for Video Coding / Streaming

The idea of preferentially allocating more resources to a region of interest during video encoding is not new [18, 19, 38]. While our primary interest is to use ROI-based bit allocation as a demonstration of the applicability of eye gaze prediction, the availability of real-time eye-gaze information does provide a firm basis for determination of ROI. In contrast, prior research without eye gaze information has to rely solely on video analysis such as high frequency content [38] and motion content [18], with the aforementioned saliency map also a suitable candidate. Nevertheless, it has been shown [39, 40] that prior knowledge and context play important roles in affecting viewer's attention. Thus, video analysis can at best provide a rough estimate of where viewers may look, in the absence of real-time information.

In contrast, we use saliency maps of video content to train HMM parameters during offline analysis, but combine real-time eye tracking information during stream time to determine ROI. The key challenge, which is the focus of this paper, is to reduce the effect of time lag due to server-client RTT delay in a networked video streaming scenario. We will show in conducted subjective testing in Section VII that ROI-based video encoding, where ROI is determined solely by saliency analysis with no real-time gaze tracking, is noticeably poorer in quality compared to video encoded in high quality for all spatial regions. On the other hand, our proposed ROI-based scheme with real-time gaze tracking performs much better in comparison.

III. Hidden Markov Model for Gaze-tracking

In this section, we discuss how we model eye gaze of a human observer watching video using a *hidden Markov model* (HMM) (section 13.2, pp.610, [41]). An HMM describes transitions of sequential state X_n's, in discrete

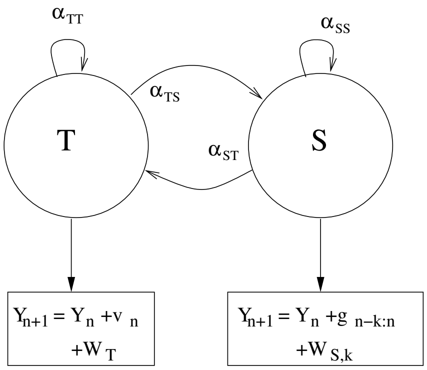


Fig. 1. Proposed hidden Markov model for eye gaze during video observation. Circles denote latent states T (tracking), which includes fixation and smooth pursuit gaze movements, and S (saccade). α's denote state transition probabilities. Y's denote the observations. v is the pixel velocity vector. g is the gaze velocity vector. W's are the additive noise terms. Boxes denote observations.

time[2] $n \in \mathcal{Z}^+$, where X_n is the *state variable* at time n. Each X_n can take on one of two possible *latent states*[3]. State T (*tracking*) models the case when the gaze of the human observer is following the motion of an identifiable object in the video. In the gaze literature [9], it is common to further categorize gaze movements into *fixation*, which models the case when eye gaze is fixated at a stationary object, and *smooth pursuit*, which models the case where gaze follows a slowly moving object. However, for our intended purpose of gaze prediction, we only need to estimate the likelihood that the human observer has identified an object of interest and is currently tracking it—doing so would mean his/her gaze location will likely coincide with the locations of the moving object in future frames as well. Thus, for simplicity we use a combined state T to model observer's tracking of object in video.

State S (*saccade*) models the rapid transition of observer's gaze from one object of interest to another. More precisely, for the purpose of gaze prediction, we interpret state S simply to mean gaze statistics that do not conform to that of tracking state T. No gaze prediction is made when state is estimated to be S due to saccade's more unpredictable nature compared to state T. Note that this definition of saccade deviates slightly from others in the literature [9], e.g., pursuit of a fast moving object (called *catch-up saccade*) will also be included in our definition of saccade. Nevertheless, this classification is more practical for our purpose of gaze prediction. Further, note that while other classifications of eye movements for the human eye are also possible [42], broadly speaking, fixation, pursuit and saccade are the three most frequently cited and major eye movement types in the literature [9].

We construct our HMM to be first-order Markovian in that the determination of state variable X_{n+1} at time $n+1$ depends solely on the value of X_n of previous time n. In particular, given $X_n = i$, the probability of $X_{n+1} = j$ is represented by *state transition probability* $\alpha_{i,j}$ of switching from state i to j. The model is hidden since the state variables X_n's are not directly observable; only observations Y_n's are observed, where each Y_n is generated by a random process dependent on current latent state $X_n = i$. The most likely state X_n given

[2] $\mathcal{Z}^+$ denotes set of positive integers.

[3] Since states $\{\mathtt{T}, \mathtt{S}\}$ cannot be observed directly, they are commonly called *latent states* in the literature.

observations $Y_1, \ldots, Y_n$ can be calculated using a simplified version of the *forward algorithm* (FA) (section 13.2.2, pp.618, [41], to be discussed). In our gaze tracking scenario, observations Y_n's can be either x- or y-coordinates of tracked eye-gaze locations on the display terminal; for simplicity, we construct the same HMM (but possibly with different parameters) to model gaze movements in x- and y-coordinates separately[4]. How we reconcile the two models during gaze prediction is discussed in Section V. Determining the most likely state X_n means, given observed tracked gaze locations, identifying the most likely eye movement type between `T` and `S`. We describe the two random processes, corresponding to latent states `T` and `S`, that generate observations next.

A. Tracking: following the motion of an identifiable object

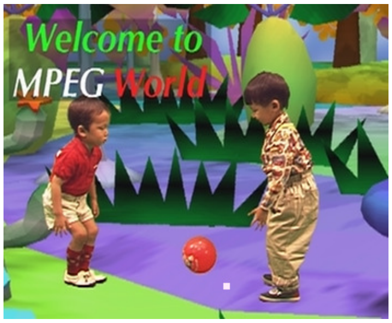


Fig. 2. An unreliable eye gaze tracker often produces noisy observations. In this example, a viewer has focused on the red ball in this frame 220 of MPEG test sequence `kids`, but an eye tracker reports gaze location marked by the 5×5 white square.

If the value of state variable X_{n+1} is `T` (tracking) at time $n+1$, we model the emitted observation Y_{n+1} as the sum of previous observation Y_n plus a *pixel velocity vector*[5] $v_n(Y_n)$, plus random noise W_T:

$$Y_{n+1} = Y_n + v_n(Y_n) + W_T \tag{1}$$

$v_n(Y_n)$ is the velocity vector of the viewed pixel, as indicated by gaze point Y_n, from frame F_n of time n to frame F_{n+1} of time $n+1$, and W_T is a zero-mean Gaussian random variable with variance σ_T^2. If the gaze point of the observer in frame F_n is known precisely, $v_n(Y_n)$ can be estimated straightforwardly via video content analysis. For example, one can use optical flow algorithms [43], or more computation-efficient block search commonly used in video coding standards like H.263 [44], H.264 [45]: first identify the macroblock that contains the viewed pixel at time n, then find the best matched macroblock in frame F_{n+1} in terms of RGB pixel values, and calculate the

[4] While it is possible to construct a single HMM to jointly consider both x- and y-coordinates, we choose to construct separate HMMs for x- and y-coordinates for two reasons: i) a simpler model requires fewer data samples for the few model parameters to converge, and ii) a simpler model has lower complexity (our gaze prediction algorithm must be executed in real-time).

[5] While pixel velocity v_n can also be considered as an observation, it is essentially a derivative of observation Y_n—movement of observed pixel located at Y_n from frame F_n to F_{n+1}. Thus we will only write Y_n as the sole independent observation value for each instant n.

corresponding motion vector. The probability of observing Y_{n+1} (*emission probability*) given current state is $\mathtt{T}$ is hence:

$$P_T(Y_{n+1}|Y_n) = f_{\sigma_T^2}(Y_{n+1} - Y_n - v_n(Y_n)) \tag{2}$$

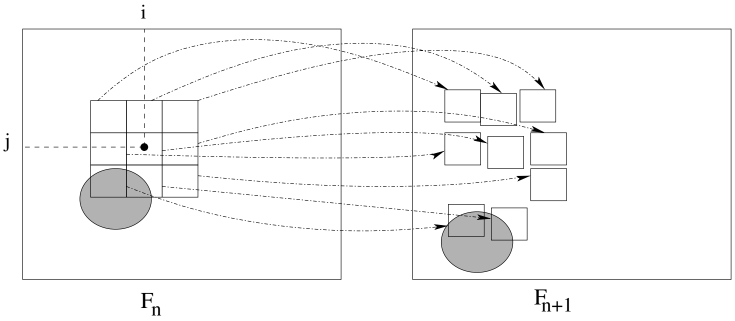


Fig. 3. Calculation of forward motion vector candidates in next frame F_{n+1} given eye gaze data Y_n at location (i, j) in frame F_n.

Unfortunately, the problem with (2) is that the true gaze point in frame F_n is not known precisely due to noise in the observation. That means that if a viewer is actually following a moving object but gaze point is not on the object due to noise (as shown in Fig. 2), then the calculated motion vector $v_n(Y_n)$ will be erroneous.

To circumvent this problem, we perform multi-block search as shown in Fig. 3. For given observed gaze location Y_n, we first identify a *neighborhood* of macroblocks around Y_n. For each macroblock in the neighborhood, we search for a best matched block in the next frame F_{n+1} and calculate the corresponding motion vector v_n. Among all the calculated vectors v_n's, we identify the one that gives the largest conditional probability for state $\mathtt{T}$:

$$\begin{aligned} P_T(Y_{n+1}|Y_n) &= \max_{v_n \in \mathcal{V}_n(Y_n)} f_{\sigma_T^2}(Y_{n+1} - Y_n - v_n) \\ v_n^* &= \arg \max_{v_n \in \mathcal{V}_n(Y_n)} f_{\sigma_T^2}(Y_{n+1} - Y_n - v_n) \end{aligned} \tag{3}$$

where $\mathcal{V}_n(Y_n)$ is the set of calculated motion vectors for a neighborhood of macroblocks around detected gaze point Y_n, and v_n^* is the motion vector in $\mathcal{V}_n(Y_n)$ that maximizes the tracking emission probability $P_T(Y_{n+1}|Y_n)$.

B. Saccade: switching fixation points

If the viewer is in state $X_{n+1} = \mathtt{S}$ (saccade) at time $n+1$, the gaze of the viewer is not following an identifiable object in the video, and thus is very likely switching from one object of interest to another. The transition process usually lasts a short duration (20 to 200ms), and the movement is fast [9]—saccade is said to be the fastest movement by the human body [9]. Fortunately, very often movement of the eye during one saccade is along a straight line [9]. Thus, if we are able to establish a *gaze vector* $g_{n-h+1:n}$ during saccade using previous observations Y_n's, then new observation Y_{n+1} is previous observation Y_n plus $g_{n-h+1:n}$, plus a noise term $W_{S,h}$.

Mathematically, we write observation Y_{n+1} given viewer resides in state $X_{n+1} = \texttt{S}$ as follows[6]:

$$Y_{n+1} = Y_n + g_{n-h+1:n} + W_{S,h} \tag{4}$$

where $g_{n-h+1:n}$ is the mean eye gaze vector computed using most recent $h \geq 2$ observations $Y_{n-h+1}, \ldots, Y_n$. $W_{S,h}$ is a zero-mean Gaussian variable, whose variance $\sigma^2_{S,h}$ depends on the number of observations, h, used to compute $g_{n-h+1:n}$. The idea is to capture the notion that, in general, the more recent observations Y_n's we use to estimate gaze vector $g_{n-h+1:n}$, the smaller the corresponding variance $\sigma^2_{S,h}$ of Gaussian noise $W_{S,h}$ should be. $g_{n-h+1:n}$ can be computed using samples $(n-h+1, Y_{n-h+1}), \ldots, (n, Y_n)$ via linear regression (section 3.1, pp.138, [41]). On the other hand, if gaze movement does not follow a straight line but a curvature instead (again, in rare cases), then more samples do not lead to better estimate of gaze vector $g_{n-h+1:n}$. In practice, we cap the maximum number of samples used to be no larger than a parameter H ($H = 15$ is used in our experiments).

We can now write the emission probability $P_S(Y_{n+1}|Y_n, \ldots, Y_{n-h+1})$ of observing Y_{n+1} given previous h observations $Y_n, \ldots, Y_{n-h+1}$ and current state is $\texttt{S}$ as follows:

$$P_S(Y_{n+1}|Y_n, \ldots, Y_{n-h+1}) = f_{\sigma^2_{S,h}}(Y_{n+1} - Y_n - g_{n-h+1:n}) \tag{5}$$

We notice that $P_T(Y_{n+1}|Y_n)$ in (3) and $P_S(Y_{n+1}|Y_n, \ldots, Y_{n-h+1})$ have similar forms and would evaluate to have similar values if v^*_n and $g_{n-h+1:n}$ are similar (if the corresponding variance σ^2_T and $\sigma^2_{s,h}$ are also similar). This is the case when the observer is tracking an object in the video with slow linear motion, so that the motion vector and gaze vector coincide. Clearly, we should label this case as tracking state $\texttt{T}$, indicating that we can predict future gaze location with high probability. To disambiguate state $\texttt{S}$ from $\texttt{T}$ in this case, we do the following: we add a weighting parameter $1 - e^{\gamma|v^*_n - g_{n-h+1:n}|}$ to probability $f_{\sigma^2_{S,h}}$, so that if motion vector v^*_n is close to gaze vector $g_{n-h+1:n}$, then emission probability $P_S(Y_{n+1}|Y_n, \ldots, Y_{n-h+1})$ is small. To summarize, we can replace the earlier (5) with the following:

$$P_S(Y_{n+1}|Y_n, \ldots, Y_{n-h+1}) = (1 - e^{\gamma|v^*_n - g_{n-h+1:n}|})\ f_{\sigma^2_{S,h}}(Y_{n+1} - Y_n - g_{n-h+1:n}) \tag{6}$$

where γ is a parameter to control the weight factor (γ is set to -0.25 in our scheme).

C. Finding most likely latent states

To find latent state probability $P(X_n = j)$ given a window of observations $Y_1, \ldots, Y_n$, we derive a simplified version of the forward algorithm (FA), which is the first half of the well known forward-backward algorithm (section 13.2.2, pp.618, [41]). It is a simplified version because, unlike the general case posed in [41], we do not have future observations $Y_{n+1}, \ldots$ when estimating X_n given real-time collected observations $Y_1, \ldots, Y_n$.

[6] There are many possible ways to model the complex saccade movement; we choose the simplest linear motion model for complexity reason.

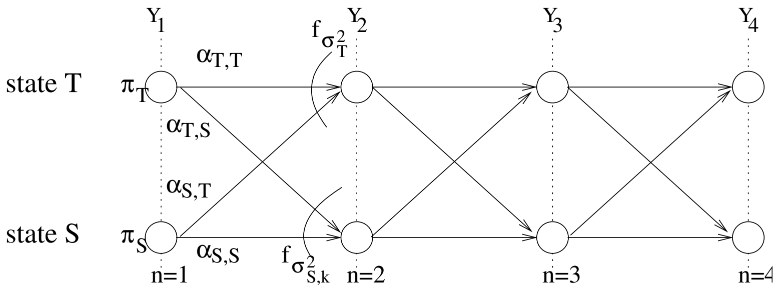


Fig. 4. Trellis corresponding to a 2-state HMM. A Forward Algorithm can find the most likely state X_n given observations $Y_1, \dots, Y_n$'s.

Mathematically, we seek to find latent variable X_n^* that maximizes the posterior probability $P(X_n|Y_1, \dots, Y_n)$ given observations $Y_1, \dots, Y_n$. Using Bayes' theorem, we can write:

$$\begin{aligned} X_n^* &= \arg\max_{X_n} P(X_n|Y_1, \dots, Y_n) \\ &= \arg\max_{X_n} \frac{P(Y_1, \dots, Y_n|X_n)P(X_n)}{P(Y_1, \dots, Y_n)} \\ &= \arg\max_{X_n} P(Y_1, \dots, Y_n, X_n) \end{aligned} \tag{7}$$

This last line follows since the choice of X_n does not affect $P(Y_1, \dots, Y_n)$. As done in [41], let $a(X_n) = P(Y_1, \dots, Y_n, X_n)$. $a(X_n)$ can be written recursively (equation (13.36), pp. 620, [41]):

$$a(X_n) = P(Y_n|X_n) \sum_{X_{n-1}} a(X_{n-1})P(X_n|X_{n-1}) \tag{8}$$

Note that (8) is computed in a recursive manner, meaning that as a new observation Y_{n+1} arrives, previously computed $a(X_n)$'s can be used for the computation of $a(X_{n+1})$'s, instead of computing the entire observation sequence $Y_1, \dots, Y_{n+1}$ again. This is equivalent to constructing a new stage of a trellis of two states representing $a(\texttt{T})$ and $a(\texttt{S})$ at instant $n+1$, reusing computed states of the previous stage at instant n, as shown in Fig. 4.

To solve (8) we still need to complete two additional practical details. First is initial conditions, which can be calculated easily as follows:

$$a(X_1) = P(Y_1, X_1) = \pi_X P(Y_1|X_1) \tag{9}$$

where π_X is the steady state probability of the Markov chain for latent state X.

The second is scaling factor. Because for each recursive call in (8) we need to multiply emission probability $P(Y_n|X_n)$ which can be much smaller than 1, $a(X_n)$ can become very small very quickly, leading to numerical instability. As done in section 13.2.4, pp.627, [41], we add in a coefficient c_n so that the sum of all $\hat{a}(X_n)$'s is 1. (8) thus becomes:

$$c_n \hat{a}(X_n) = P(Y_n|X_n) \sum_{X_{n-1}} \hat{a}(X_{n-1})P(X_n|X_{n-1}) \tag{10}$$

where c_n is chosen so that $\sum_{X_n} \hat{a}(X_n) = 1$.

IV. Analysis of Visual Saliency Maps

A. Overview of Saliency Map Analysis

For the previously presented HMM to correctly model a viewer's eye-gaze movements during playback of a video clip, model parameters (most importantly, HMM state transition probabilities) appropriate for the observed video clip must be derived. Different video contents contain different visual excitation through stimuli properties, inducing different amount of eye-gaze movements from viewers. For example, a video capturing a head-and-shoulder sequence of the president addressing the nation may induce very few gaze movements, while a dance music video with lots of new objects entering and leaving the scene may induce a lot. Thus, finding suitable HMM parameters given the visual activities of the video is important for eye-gaze movement modeling.

One brute-force method to derive appropriate HMM parameters for a given video content is to conduct extensive eye-gaze experiments [46], using a real-time gaze tracking system [20], with a sizable group of test subjects. This, however, is clearly too time-consuming and cost-ineffective for a large number of video clips.

Instead, we propose an alternative method to derive HMM parameters per video clip by analyzing the visual saliency maps [10] of individual video frames across time. Computed saliency maps for individual video frames describe visual attention variation *spatially*. For our purpose, we seek to describe visual attention variation of a video *temporally*, *i.e.*, how a viewer will shift visual attention from one object of interest to another over time, which requires additional steps.

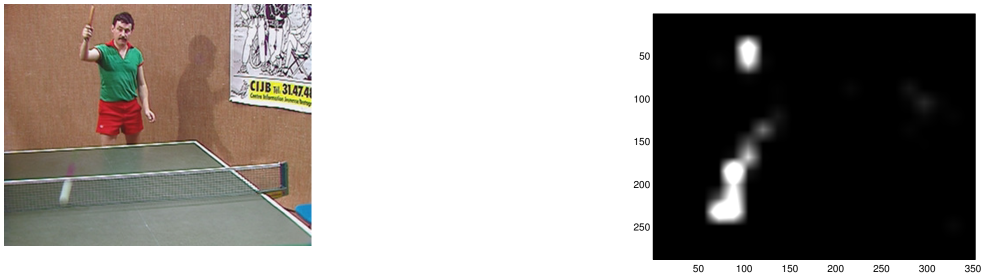
Our methodology is as follows. First, we define *saliency objects* within each video frame given calculated saliency maps; as a first-order approximation, saliency objects are the only regions a viewer may observe at that particular frame. Then, we derive HMM transition probabilities of a Markov model by solving consistency equations written for different saliency objects across consecutive frames. Finally, we can optionally segment a video into shorter clips of roughly stationary gaze statistics[7] by computing the Kullback-Leibler (KL) divergence using motion-compensated saliency maps of consecutive frames. We describe these steps in order next.

B. Identification of Saliency Objects

We first compute visual saliency maps for all video frames using methodology in [10]. As an example, in Fig. 5 we see an original video frame, frame 157 of MPEG test sequence `table`, and its corresponding computed saliency map. We see that saliency values are highest around the ping-pong ball and the hand, agreeing with our expectation of visual attention for this frame.

1) Finding Initial Saliency Objects: Having computed visual saliency maps, we first normalize each individual map, so the sum of all saliency values in a map equals to one. We then find a set of *saliency objects* in each map. We define a saliency object as a spatially connected region with per-pixel saliency value larger than a pre-defined threshold τ_s. As a first order of approximation, we assume these are the only video objects a viewer will observe

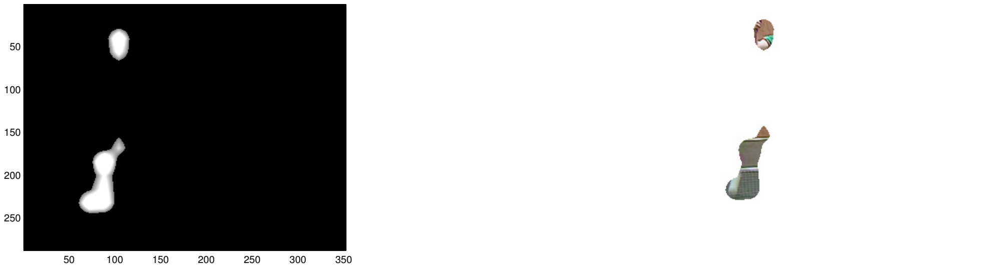
[7] By a video clip of stationary gaze statistics, we mean a set of consecutive video frames where the induced gaze movements can be modeled by the same set of HMM parameters.



(a) original video frame

(b) corresponding saliency map

Fig. 5. Original video frame 157 of MPEG test sequence `table`, and the corresponding visual saliency map, calculated using method in [10].



(a) saliency map w/ threshold

(b) saliency objects

Fig. 6. Normalized saliency map after applying threshold, and resulting salient objects in video frame 157 of sequence `table`.

in the given frame. A viewer may of course have gaze locations outside of these saliency objects; we assume that such occurrence means the viewer is in the process of switching from one saliency object to another; *i.e.*, he is in saccade state at this frame time. Returning to our earlier example, we see in Fig. 6(a) the normalized saliency map with normalized saliency values below threshold τ_s set to zero, leaving only two saliency objects in the map. Correspondingly, we see the saliency objects in Fig. 6(b).

2) Merging of Saliency Objects: Because the computed saliency maps can be noisy, it turns out that finding a single appropriate threshold τ_S *a priori* that can identify reasonable saliency objects in all saliency maps of a video in time is difficult. To ease the burden of the threshold selection, we perform the following two procedural steps after initial saliency objects are found in a frame. First, if only a single saliency object is found in a frame, we incrementally lower threshold τ_S until a second saliency object is discovered. We do so because, by definition, the probability of a viewer being in saccade state in any frame is non-zero (*i.e.*, there is a non-zero chance of a viewer switching objects of interest in any frame), and having a single saliency object means there are no other objects to switch to. Performing such procedure usually means the size of the original single saliency object increases as threshold τ_S is lowered. This agrees with intuition: the original object remains the main object with the strongest visual attention despite the decrease in threshold.

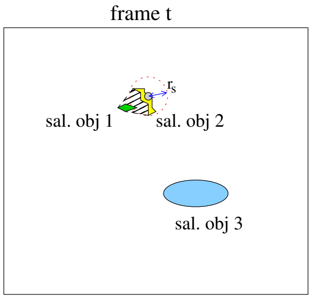


Fig. 7. Example of how two small saliency objects (obj. 1 and obj. 2) are merged using search radius r_s.

Second, for each discovered saliency object, we search within a radius r_s of the object's center[8] to check if another object is in the vicinity. If so, we merge the two objects via convex combination. In Fig. 7, we see saliency object 1 is within radius r_s of object 2's center, so we merge the two objects into one. The motivation of saliency object merging is the following. A viewer necessarily looks at a group of pixels at a time. So if a very small object $o_{t,i}$ (smaller than a circle of radius r_s) is in the vicinity of another object $o_{t,j}$, then the viewer is also looking at object $o_{t,j}$ when observing $o_{t,i}$. Thus it is sensible to merge the two objects.

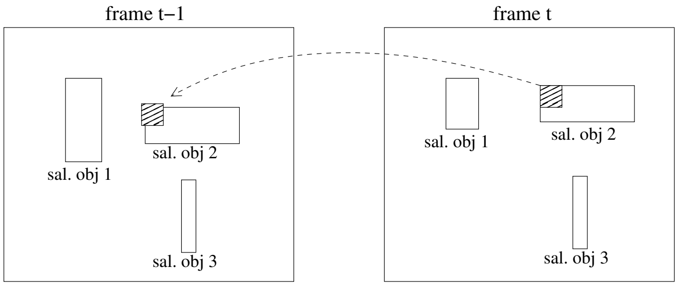


Fig. 8. Example of how correspondence of saliency objects located in pairs of consecutive saliency maps are found using motion estimation (ME).

3) Correspondence of Saliency Objects: We can establish correspondence among saliency objects in consecutive frames using motion estimation (ME), commonly used in video compression algorithms [44, 45]. In details, for each block k (we use 8×8 in our experiments) in a saliency object $o_{t,i}$ in saliency map of instant t, we find the most similar block in saliency map of instant $t-1$, *i.e.* the block with corresponding RGB pixel values in the original video frame $t-1$ that most matches RGB pixel values corresponding to block k in frame t. If the most similar block in saliency map $t-1$ belongs to a saliency object $o_{t-1,j}$, then object $o_{t-1,j}$ in map $t-1$ and object $o_{t,i}$ in map t could potentially be the same object. If a sufficiently large fraction of blocks k's in $o_{t,i}$ map to the same object in $o_{t-1,j}$, then we declare they are the same object. If no such object exists in previous map $t-1$, then we

[8] An object center (c_x, c_y) is the Cartesian center of the object, where c_x and c_y are the arithmetic means of the x- and y-coordinates of every pixels in the object.

declare object $o_{t,i}$ to be a new object appearing for the first time in map t. As an example, in Fig. 8, we see that a block in object 2 in frame t has found a matching block in object 2 in frame $t-1$.

C. Deriving Transition Probabilities

Having identified saliency objects across frames, we now derive state transition probabilities for our eye-gaze HMM. As an illustrative example, we examine the simple case where there are the minimum two salient objects in consecutive frames t and $t+1$. Denote by $p_{t,1}$ and $p_{t,2}$ the probability that a viewer will fix his gaze in each of the two objects, respectively, in frame i. Similarly, denote by $p_{t+1,1}$ and $p_{t+1,2}$ the corresponding probabilities for frame $t+1$. Let s_t and s_{t+1} be the probabilities that a viewer is in saccade state in frame t and $t+1$. Because we know the *volume* of visual saliency for each saliency object (sum of computed saliency pixel values within each object) and saccade spatial region (area *not* covered by saliency objects), we can calculate the relative probability size of objects by comparing their volumes in each frame:

$$\begin{aligned} s_t &= p_{t,1}/\beta_{t,1} = p_{t,2}/\beta_{t,2} \\ s_{t+1} &= p_{t+1,1}/\beta_{t+1,1} = p_{t+1,2}/\beta_{t+1,2} \end{aligned} \tag{11}$$

where β's are the scaling factors among objects in each frame.

Further, we know that the sum of probabilities in each frame must equal 1:

$$\begin{aligned} p_{t,1} + p_{t,2} + s_t &= 1 \\ p_{t+1,1} + p_{t+1,2} + s_{t+1} &= 1 \end{aligned} \tag{12}$$

Together with (11), we can determine the gaze probability of each object in each frame. This is true no matter how many saliency objects are in each frame.

To calculate the state transition probabilities α's, we apply the definition of state transition to the objects of these two frames. We can write the probability $p_{t+1,1}$ of object 1 in frame $t+1$ as the sum of probabilities of objects in previous frame scaled by view transition probabilities α's:

$$p_{t+1,1} = p_{t,1}\ \alpha_{TT} + s_t\ \alpha_{ST} \left(\frac{\beta_{t+1,1}}{\sum_{i=1}^{2} \beta_{t+1,i}} \right) \tag{13}$$

Note that the probability $s_t\ \alpha_{ST}$ from state $\mathtt{S}$ to $\mathtt{T}$ must be split between the two objects, according to their relative volumes.

We can write a similar equation for probabilities $p_{t+1,2}$ of moving objects 2 in frame $t+1$. Further, we can similarly write state transition equation for the saccade state as well:

$$s_{t+1} = \alpha_{TS} \sum_{i=1}^{2} p_{t,i} + s_t\ \alpha_{SS} \tag{14}$$

Note that we have now three state transition equations for the four unknown α's. In general, one can obtain $k+1$ state transition equations for k saliency objects. In addition, we know the sum of probabilities leaving a state in a

HMM must also be one:

$$\begin{aligned} \alpha_{TT} + \alpha_{TS} &= 1 \\ \alpha_{SS} + \alpha_{ST} &= 1 \end{aligned} \tag{15}$$

These two linear equations, together with the earlier derived three linear state transition equations, means that we have more equations than unknowns. We hence compute α's as follows. We rewrite each linear equation i with an additional noise term n_i at the end. The set of linear equations becomes:

$$C\mathbf{a} = \mathbf{b} + \mathbf{n} \tag{16}$$

where $\mathbf{a} = [\alpha_{TT}, \alpha_{TS}, \alpha_{SS}, \alpha_{ST}]^T$ is the vector of α's we are seeking, C is the coefficient matrix, $\mathbf{b}$ and $\mathbf{n}$ are the constant and noise vectors, respectively. It is well known that the $\mathbf{a}^*$ that minimizes the noises $\mathbf{n}$ in a mean square sense is computed as follows:

$$\mathbf{a}^* = C^+\mathbf{b} \tag{17}$$

where $C^+ = (C^TC)^{-1}C^T$ is the Moore-Penrose pseudo-inverse of matrix C.

Having computed sets of transition probabilities α's each using different pairs of neighboring saliency maps in time, the transition probabilities for the video is simply the average of the computed sets of transition probabilities. We can then also compute the steady state probabilities π's of the HMM by performing eigen-analysis as typically done in the literature.

D. Partitioning Video into Stationary Segments

For better performance, video can be partitioned into segments of roughly stationary gaze statistics, so that segment-specific HMM parameters can be used. While there are several methods in literature to divide video into segments corresponding to stationary content, our purpose requires an alternative approach. The segmentation should not rely directly on the analysis of the video content itself but on the visual saliency instead. For instance, if two different video shots have similar saliency characteristics, then there is no reason to use two different set of HMM parameters. Consequently, we propose to track how fast gaze statistics are susceptible to change in the video based on saliency map analysis in time.

We first compute *motion-compensated saliency maps*: after identifying saliency objects in saliency map t and $t+1$, for each corresponding saliency object pair in map t and $t+1$, we relocate the object in map $t+1$ to match the location of the corresponding object in map t. Such relocation process allows easier comparison of saliency characteristics frame-to-frame in terms of gaze statistics, particularly when a salient object is in motion.

The comparison is achieved treating saliency map ϕ_t at t and motion-compensated saliency map ϕ_{t+1} at $t+1$ as probability distribution functions, and compute the Kullback-Leibler (KL) divergence as follows:

$$d_{KL}(\phi_t||\phi_{t+1}) = \sum_i \phi_t(i) \log\left(\frac{\phi_t(i)}{\phi_{t+1}(i)}\right) \tag{18}$$

If the computed KL divergence exceeds a certain threshold τ_{KL}, then we declare there is an abrupt change in statistics, and we can partition the video clip into two segments of roughly stationary gaze statistics.

V. Gaze Prediction using Kalman Filter

We have discussed how to determine the most probable latent state X_n in HMM given observations $Y_1, \ldots, Y_n$ in Section III. In this section, we discuss how a future gaze location $\hat{Y}_{n+RTT}$ can be estimated RTT gaze samples into the future. Smart bit allocation can then be performed to assign finer quantization parameter (QP) for ROI centered on predicted location $\hat{Y}_{n+RTT}$, and coarser QP for other spatial regions in a coded frame (to be discussed in Section VI).

We stress here that we perform gaze prediction *only if* the most likely state is $\texttt{T}$. This may seem counter-intuitive, since it is commonly accepted that the human eyes cannot perceive any visual details when in saccade state $\texttt{S}$ [42], and so it appears that, for the ROI bit allocation application, the greatest bit-saving can be achieved when the viewer is in state $\texttt{S}$. However, the duration in which a viewer stays in state $\texttt{S}$ is typically very short [9], and gaze will soon stop at an unpredictable new object of interest. Thus, reducing bit-rate through coarser quantization of the video frames when viewer is in state $\texttt{S}$ poses a significant risk of not reacting fast enough to improve video quality back up when viewer suddenly switches from state $\texttt{S}$ to $\texttt{T}$. This is particularly the case when a low-cost web camera capturing video at a low frame rate is used for gaze tracking. Hence, we take the conservative approach and perform no gaze prediction in state $\texttt{S}$.

Further, even if the most likely state is $\texttt{T}$, we perform prediction only if probability $P(X_n = \texttt{T}|Y_1, \ldots, Y_n)(\alpha_{TT})^{RTT}$ exceeding a threshold τ_C for both x- and y-coordinate state estimation. In other words, we employ prediction of gaze location to perform optimized bit-allocation *only if*:

1) We have confidence in our state estimation $P(X_n = \texttt{T}|Y_1, \ldots, Y_n)$; and,
2) The likelihood of the observer staying in state $\texttt{T}$ RTT gaze samples into the future remains high.

For example, a long RTT between server and client, or a video content that contains many salient objects and induces much gaze movement (small α_{TT}), will limit the fraction of time we actually make gaze prediction.

A. Gaze Data Denoising using Kalman Filtering

Given $P(X_n = \texttt{T}|Y_1, \ldots, Y_n)(\alpha_{TT})^{RTT} \geq \tau_C$, we first denoise D latest samples of noise-corrupted observations $Y_{n-D+1}, \ldots, Y_n$ into estimated gaze points $\hat{Y}_{n-D+1}, \ldots, \hat{Y}_n$ using *Kalman filtering* (KF) [13]. D is the size of a small window of previous gaze samples (for complexity reason) that have been estimated to be state $\texttt{T}$ during HMM state estimation. To conform to the standard KF formulation, we modify previous notation to the following. Let $\hat{Y}_n$ and $\dot{\hat{Y}}_n$ be the true gaze location and velocity at time n. Denote by $\hat{\mathbf{Y}}_n = [\hat{Y}_n \; \dot{\hat{Y}}_n]^T$ the 2×1 vector that contains the true gaze location and velocity at time n. We write the evolution of $\hat{\mathbf{Y}}_n$ recursively as a *linear dynamic system* (LDS):

$$\hat{\mathbf{Y}}_n = \underbrace{\begin{bmatrix} 1 & (1-\beta) \\ 0 & (1-\beta) \end{bmatrix}}_{\mathbf{F}_n} \hat{\mathbf{Y}}_{n-1} + \underbrace{\begin{bmatrix} \beta \\ \beta \end{bmatrix}}_{\mathbf{B}_n} v^*_{n-1} + \begin{bmatrix} W_P \\ 0 \end{bmatrix} \tag{19}$$

where $\mathbf{F}_n$ and $\mathbf{B}_n$ are respectively the state transition and control-input models, v^*_{n-1} is the control vector (also the emission probability maximizing block motion vector in (3)), and W_P is a zero-mean Gaussian process noise with variance σ_P^2 . In words, (19) states that the next true gaze location $\hat{Y}_n$ is the previous gaze location $\hat{Y}_{n-1}$, plus $(1-\beta)$ times the gaze vector $\dot{\hat{Y}}_{n-1}$, plus β times the maximizing block motion vector v^*_{n-1}, plus a noise term W_P. β is a parameter to control the convex combination of previous gaze velocity vector and motion vector of the scene. In our experiments, β is set close to 1. Note that having first derived v^*_{n-1} using (3) during HMM state estimation in Section III, it is then possible to write (19) as a LDS in each given instant n.

The observation $\mathbf{Y}_n = [Y_n \ \dot{Y}_n]^T$ is simply $\hat{\mathbf{Y}}_n$ plus an observation noise term:

$$\mathbf{Y}_n = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \hat{\mathbf{Y}}_n + \begin{bmatrix} W_O \\ 0 \end{bmatrix} \tag{20}$$

where W_O is a zero-mean Gaussian observation noise with variance σ_O^2.

Having written the evolution and observation equation (19) and (20), we can compute the estimated gaze location $\hat{Y}_{n-D+1}, \ldots, \hat{Y}_n$ using standard KF predict and update equations. See [13] for details.

B. Gaze Prediction using Linear Dynamic System

Given estimated gaze point $\hat{Y}_n$, we predict gaze RTT samples into the future using a similar LDS setup. However, because there are no future observations available beyond Y_n, Kalman filtering reduces to a simpler LDS setup with no data denoising. We write a similar evolution equation for $\mathbf{Y}_n$ into the future as follows:

$$\mathbf{Y}_n = \underbrace{\begin{bmatrix} 1 & (1-\beta) \\ 0 & (1-\beta) \end{bmatrix}}_{\mathbf{F}_n} \mathbf{Y}_{n-1} \; + \; \underbrace{\begin{bmatrix} \beta c_1 & \ldots & \beta c_Z \\ \beta c_1 & \ldots & \beta c_Z \end{bmatrix}}_{\mathbf{B}_n} \underbrace{\begin{bmatrix} v^1_{n-1} \\ \vdots \\ v^Z_{n-1} \end{bmatrix}}_{\mathbf{u}_n} \tag{21}$$

where $v^1_{n-1}, \ldots, v^Z_{n-1}$ are the Z block MVs around gaze point Y_{n-1}.

In words, (21) states that gaze location Y_n is the previous location Y_{n-1} plus $(1-\beta)$ times previous velocity $\dot{Y}_{n-1}$, plus β times a weighted combination of MVs of the surrounding blocks $v^1_{n-1}, \ldots, v^Z_{n-1}$, where weights c_i's sum to 1, $\sum_{i=1}^{Z} c_i = 1$. The weights $c_1, \ldots, c_Z$ are used to compensate for the fact that observation Y_{n-1} is not available to select the MV that maximizes the emission probability, as done in (3). To predict gaze location RTT samples into the future, we repeatedly compute (21), starting from the last estimated gaze location $\hat{Y}_n$. After RTT iterations, we have an estimated gaze location $\hat{Y}_{n+RTT}$ into the future.

VI. ROI Bit Allocation for Video Encoding

In this section, we discuss a bit-rate allocation strategy as an application of our proposed HMM based eye-gaze prediction method. Conceptually, human ability to appreciate pixel fidelity decreases continuously away from the center of focus. Hence it is wasteful to encode visual information away from focus with high fidelity. In the previous sections, we already described how to predict the location of future eye gaze $\hat{Y}_{n+RTT}$. One approach to exploit this knowledge of user's visual focus is to continuously adapt each macroblock's quantization parameter (QP) according to a visual model [19]. Nevertheless, in this paper, we adopt a simpler approach in which a rectangular ROI is determined, and one QP is assigned to the ROI, while a coarser (higher) QP is assigned to spatial regions outside the ROI. This is due to its lower complexity, and the lower sensitivity to errors in focus determination. Specifically, regions far away from focus is no longer subjected to extreme quantization, which yields little additional rate reduction, but may attract unwanted attention due to large quantization artifacts, changing the visual saliency of the original video frames [47].

A. Bit Allocation of ROI

As discussed in [17], the fall-off in human ability to appreciate pixel fidelity can be approximately modeled by the contrast sensitivity (CS) of humans, which is the reciprocal of the contrast threshold (CT) given by:

$$\begin{aligned} CT(f,e) &= CT_0 \exp\big(\alpha f \frac{e+e_2}{e_2}\big) \\ CS(f,e) &= 1/CT(f,e) \end{aligned}$$

where f is spatial frequency, e is the retinal eccentricity or the angle relative to the point of focus, and CT_0, e_2 and α are constants empirically determined to be 1/64, 2.3, and 0.106, respectively.

As done in [19], we determine the cutoff frequency, f_c, by setting CT to one:

$$f_c = \frac{e_2 \log \frac{1}{CT_0}}{\alpha(e_{\max} + e_2)} \tag{22}$$

where $e_{\max}$ is the maximum eccentricity in the video frame, which is the largest angle the screen portends relative to the focus point. The average contrast threshold evaluated at spatial frequency f_c inside and outside an ROI are then computed, and the corresponding QP are chosen so that:

$$\frac{QP_{ROI}}{CT_{ROI}} = \frac{QP_{\overline{ROI}}}{CT_{\overline{ROI}}} \tag{23}$$

For ease of computation, we are primarily interested in having only two regions, namely inside and outside the ROI, and having rectangular ROI. Nevertheless, the scheme can be trivially extended to multiple regions, and to non-rectangular ROI.

In addition, the saliency map will change corresponding to the QP change. To avoid the effect, QP also should be selected carefully according to:

$$D_{KL}(QP_{ROI} + QP_{\overline{ROI}} || QP_{Full}) < \varrho \tag{24}$$

B. Determining ROI for State T

Given a video frame with width w and height h, we choose a ROI of size $w/2 \times h/2$ centered at the estimated gaze location. This allows at least 75% of the frame to be coded at a lower QP, while allowing a substantial region near the focus point to be at high quality. For experiments in Section VII with a field of view of 55 degrees, this corresponds to a ROI with field of view of 30 degrees, which is quite large to comfortably capture regions of high visual sensitivity.

C. Confirmation of ROI using Saliency Map Analysis

To ensure that the predicted observer's gaze movement does synchronize with an identified moving object in the video from frame F_n to F_{n+RTT}, we perform one final check to see if the predicted gaze location $\hat{Y}_{n+RTT}$ lands inside the same saliency object in frame F_{n+RTT} as it did in frame F_n. If it does not, then we declare uncertainty in the prediction, and the entire frame is encoded in high quality.

VII. Experimentation

We demonstrate the benefit of our proposed HMM-based gaze prediction strategy through both objective and subjective experiments. We first describe the setup of our experiments in Section VII-A. In part one of the experiment in Section VII-B, we show that our proposed saliency map analysis can be used to derive accurate HMM parameters. In part two, described in Section VII-C, we examine the accuracy of our HMM state estimation, and the tradeoff between false positive (predicting HMM state to be T when ground truth is S) and false negative (predicting HMM state to be S when ground truth is T). In part three, described in Section VII-D, we examine the accuracy of our HMM-based gaze prediction using Kalman filtering. In part four, described in Section VII-E, we examine the achievable bit-rate saving for our proposed bit allocation scheme. Finally, through an extensive subjective study, we show that our bit allocation scheme suffers no statistically meaningful loss in perceived visual quality, using our in-house developed real-time system, in Section VII-F.

A. Experimental Setup

Our gaze-based networked streaming system employs the free real-time gaze-tracking software opengazer [20], which is calibrated for sampling gaze location at 30 samples per second using an off-the-shelf web camera. In our experiment, we used two kinds of sequences: i) 300-frame standard MPEG video test sequences at CIF resolution (352 × 288), and ii) 150-frame video sequence at SD resolution (720 × 576) that can be downloaded from [48]. To mitigate viewer frustration from repetitive viewing, we used five CIF videos: *silent, table, mother, foreman, kids*, and five SD videos: *captain, group, racing, rowboat, concert.*

The monitor used for gaze tracking and video experiments measured 24 inches diagonally ($522.3mm \times 329.6mm$), with resolution of 1920×1200. Brightness and contrast are set to 30% and 50%, respectively. The distance between a user's head and the center of monitor screen is about $500mm$, resulting in a viewing angle of about 55 degrees to the side-edge of the screen.

For video compression, we use a fast implementation of H.263 [44] for real-time encoding. For subjective testing, videos were displayed in full-screen mode at 30 fps (for CIF) and 15 fps (for SD), either the same or half the sampling rate of opengazer for one-to-one or two-to-one correspondence between gaze samples and video frames.

B. Validation of Saliency Map Analysis for HMM Parameter Derivation

We now validate our proposed saliency map analysis discussed in Section IV, *i.e.*, whether HMM state transition probabilities derived from saliency map analysis are roughly the same as "ground truth data". The ground truth model parameters are derived as follows. First, a trained user performed multiple viewings of each test sequence, each time recorded his intention of tracking state `T` or saccade state `S` by pressing keys on a keyboard during state transitions. This data set serves as initial guess Θ of the ground truth HMM model parameters.

Then, we use the forward-backward algorithm (section 13.2.2, pp.618, [41]) to refine model parameters Θ as follows. In Section III-C, we defined forward probability $a(X_n)$ in (8). We now define its counterpart—backward probability—as follows (equation (13.38), pp.622, [41]):

$$\begin{aligned} b(X_n) &= P(Y_{n+1}, \ldots, Y_N | X_n) \\ &= \sum_{X_{n+1}} b(X_{n+1}) P(Y_{n+1}|X_{n+1}) P(X_{n+1}|X_n) \end{aligned} \tag{25}$$

where X_n is the latent state at instant n, and Y_n is the observed gaze location at instant n, $n \in \{1, \ldots, N\}$. Like (8), (25) can also be computed recursively.

Using forward probability $a(X_n)$ and backward probability $b(X_n)$, we can calculate the following quantity (equation (13.43), pp.623, [41]):

$$\xi(X_{n-1}, X_n) = \frac{a(X_{n-1}) P(Y_n|X_n) P(X_n|X_{n-1}) b(X_n)}{P(Y)} \tag{26}$$

Finally, we can estimate transition probability $\alpha_{j,k}$ from state j to k using $\xi(X_{n-1}, X_n)$ (equation (13.19), pp.617, [41]):

$$\alpha_{j,k} = \frac{\sum_{n=2}^{N} \xi(X_{n-1} = j, X_n = k)}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(X_{n-1} = j, X_n = l)} \tag{27}$$

where l takes on all possible latent state values, which in our case is simply state `T` and `S`.

HMM parameters can be calculated by repeating the above equations until the differences of the HMM parameters between iterations are all lower than a pre-set threshold ς = 1e-05. We use the resulting HMM model parameters as "ground truth data".

State transition and steady state probabilities for `silent` and `table` are shown in Table I(a) and II(a), respectively. Notice that `silent` is a relatively "quiet" video [49]—one with little visual attention shifts, with the saccade steady state probability π_S much smaller than `table`. For comparison, the state transition probabilities derived via our proposed visual saliency map analysis for `silent` and `table` are shown in Table I(b) and II(b), respectively. We see that the derived HMM parameters using saliency maps analysis are fairly close to the ground truth gaze data trace. In particular, we see that the analytical saccade steady state probability π_S for both `silent` and `table` are very close to the ground truth trace numbers, even though π_S's for `silent` and `table` are very different. This shows accuracy of our proposed saliency map analysis.

We performed the same experiment for the two SD test sequences, `captain` and `group` as well. `captain` is a "quiet" video, while `group` is a "busy" video. The resulting HMM state transition and steady state probabilities

TABLE I

State transition and steady state probabilities for SILENT

(a) Forward-Backward algorithm

	T	S	π
T	0.891	0.109	0.841
S	0.577	0.423	0.159

(b) saliency map analysis

	T	S	π
T	0.885	0.115	0.836
S	0.588	0.412	0.164

TABLE II

State transition and steady state probabilities for TABLE

(a) Forward-Backward algorithm

	T	S	π
T	0.893	0.107	0.598
S	0.159	0.841	0.402

(b) saliency map analysis

	T	S	π
T	0.865	0.135	0.546
S	0.162	0.838	0.454

are shown in Table III and IV. We again see very similar numbers between HMM parameters derived using saliency map analysis and ones obtained using eye-gaze data trace. Having validated our approach, we will henceforth use HMM parameters derived from saliency map analysis.

We next illustrate through examples how video can be partitioned into segments of roughly stationary gaze statistics using computed KL divergence of motion-compensated saliency maps, as discussed in Section IV-D. For our illustration, we constructed two composite video clips. The first CIF video clip consists of 100-frame of

TABLE III

State transition and steady state probabilities for CAPTAIN

(a) Forward-Backward algorithm

	T	S	π
T	0.882	0.118	0.699
S	0.274	0.726	0.301

(b) saliency map analysis

	T	S	π
T	0.924	0.076	0.643
S	0.137	0.863	0.357

TABLE IV

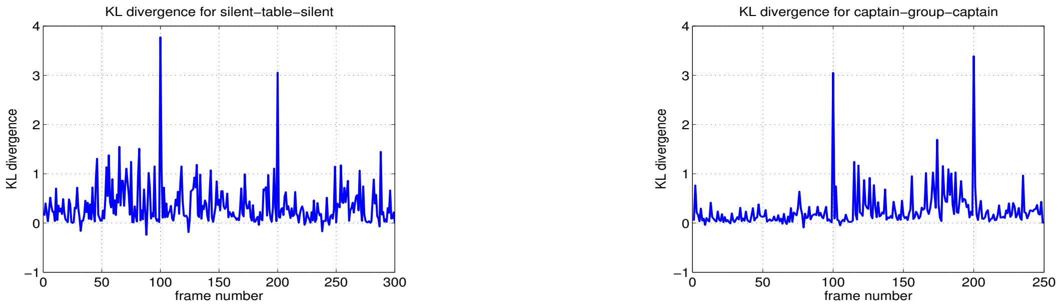
State transition and steady state probabilities for GROUP

(a) Forward-Backward algorithm

	T	S	π
T	0.823	0.177	0.356
S	0.122	0.878	0.644

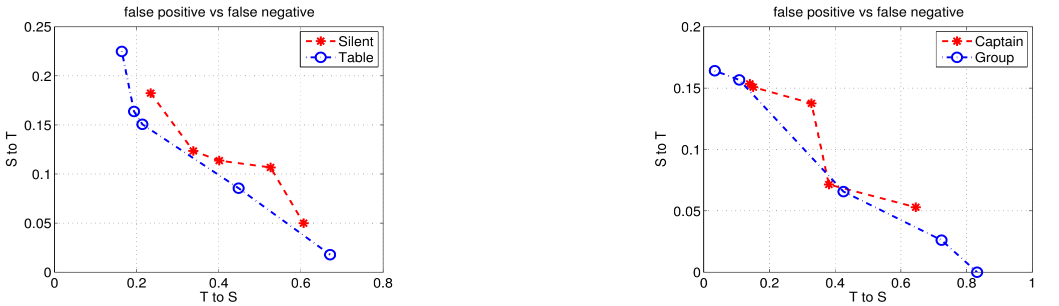
(b) saliency map analysis

	T	S	π
T	0.879	0.121	0.367
S	0.067	0.933	0.633



(a) KL divergence for `silent-table-silent` (b) KL divergence for `captain-group-captain`

Fig. 9. KL Divergence as function of frame numbers



(a) Error tradeoff for CIF videos (b) Error tradeoff for SD videos

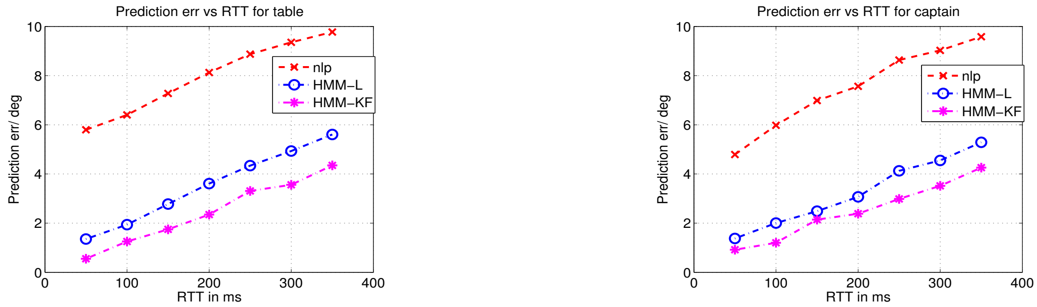
Fig. 10. Tradeoff in false positive and false negative probabilities by adjusting threshold τ_C, for CIF and SD videos, respectively.

`silent`, plus 100-frame of `table`, plus 100-frame of `silent`. Since we know the visual activities in `silent` and `table` are very different, we know *a priori* that there is a change in gaze statistics at frame 101 and 201. Similarly, we constructed a second composite SD video clip consisting of 100-frame of *captain*, plus 100-frame of `group`, plus 50-frame of `captain`.

The computed KL divergence for each frame is shown in Fig. 9(a) and (b), respectively, for the two composite sequences. We can clearly see spikes around composition frames 101 and 201, indicating a significant change in gaze statistics. This suggests that KL divergence using motion-compensated saliency maps can be an effective method to partition video into segments of different gaze statistics (even though other methods may also be appropriate).

C. Results for HMM State Estimation

We now evaluate the accuracy of HMM state estimation using forward algorithm (FA), as discussed in Section III-C. We denote an occurrence as *false positive* when FA estimates HMM state to be `T` but the ground truth state is `S` (`S` to `T`). In other words, false positive is when we wrongly deduced an opportunity to save coding bits by assigning coarser quantization parameter outside ROI, but the algorithm calls for high quality encoding for the entire frame. In contrast, we denote an occurrence as *false negative* when FA estimates HMM state to be `S` but



(a) Prediction error vs. RTT for `table`

(b) Prediction error vs. RTT for `captain`

Fig. 11. Prediction Error in degree as function of RTT for different prediction schemes, for `table` and `captain`, respectively.

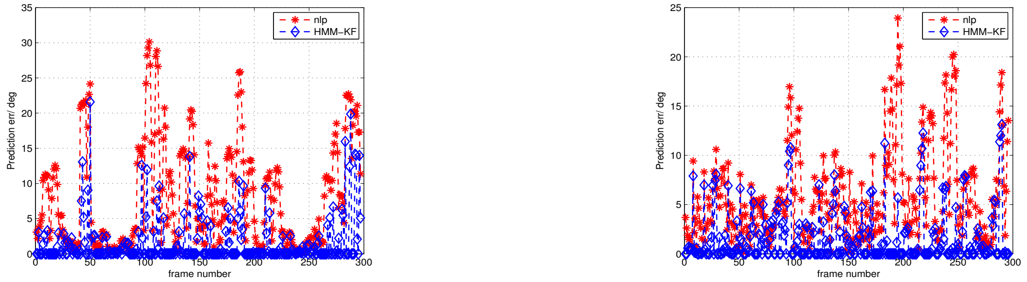
ground truth state is `T` (`T` to `S`). This is the case where we miss a bit-saving opportunity.

As discussed in Section V, a threshold τ_C can be adjusted according to our confidence in the estimated `T` state, resulting in a tradeoff between false positive and false negative probabilities. In Fig. 10, we see the said tradeoff in the two probabilities in our HMM state estimation for the two CIF (`silent` and `table`) and SD (`captain` and `group`) sequences, respectively. We see that though in general it is difficult to achieve very small false positive and false negative probabilities at the same time, it is possible to have reasonably small (≤ 0.15 for false positive and ≤ 0.2 for false negative) values for both. This shows that FA can provide reasonable state estimates for our proposed HMM. As we will discuss later, this level of estimation accuracy is sufficient for our intended application of ROI-based bit allocation for streaming video.

D. Results for Kalman Filter Prediction

Given estimated HMM states, we next examine the accuracy of our HMM-based gaze prediction using Kalman filter (`HMM-KF`), as discussed in Section V. We compare first `HMM-KF` to a naïve linear prediction scheme (`nlp`), where the last two gaze data points are used to construct a straight line, which is then extrapolated to RTT seconds later to yield a gaze location estimate. We also compare `HMM-KF` to our previous HMM-based linear prediction scheme (`HMM-LP`) [46], where linear regression is used to construct a straight line using a window of previous gaze samples, then extrapolated into the future for gaze estimate as done for `nlp`. In Fig. 11, we see the performance of all schemes, in terms of visual degree between the estimated gaze locations and true gaze locations, as function of RTT for CIF sequence `table` and SD sequence `captain`. We see that as RTT increased, the estimation error increased for all schemes. However, `HMM-LP` and `HMM-KF` achieved much smaller errors than `nlp`. This is because, to contain errors, `HMM-LP` and `HMM-KF` construct a prediction only when they are sufficiently confident that the viewer's gaze is in tracking state `T`, while `nlp` makes an estimate for all data points.

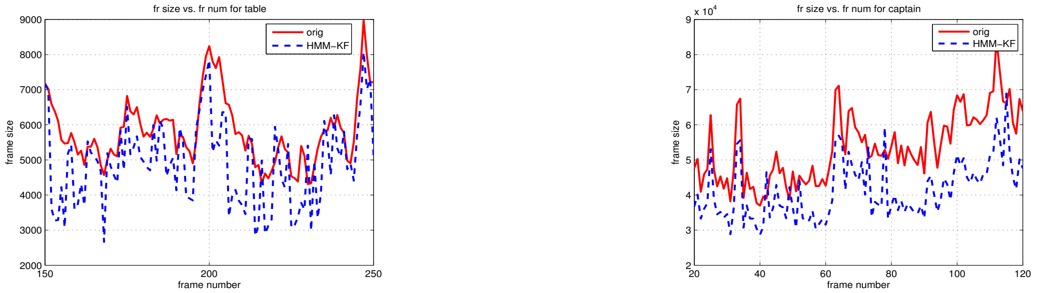
Second, we observe that `HMM-KF` performed better than `HMM-LP`. This is because the linear dynamic system employed in `HMM-KF` is able to deduce the true motion of an identifiable object in future video, while `HMM-LP` simply assumes linear motion.



(a) Prediction error vs. frame number for `table`

(b) Prediction error vs. frame number for `captain`

Fig. 12. Prediction Error in degree as function of frame number for different prediction schemes, for `table` and `captain`, respectively, when RTT=200ms.



(a) Frame size vs. frame number for `table`

(b) Frame size vs. frame number for `captain`

Fig. 13. Frame size as function of frame number for different bit allocation schemes, for `table` and `captain`, respectively, when RTT=200ms.

We also plotted the resulting prediction error in Fig. 12 against frame number for $RTT = 200ms$ for `HMM-KF` and `nlp`. At frame numbers where `HMM-KF` made prediction, we observe that the magnitude of resulting error was in general smaller than `nlp`.

E. Results for HMM-based Bit Allocation

We next show the achievable bit saving for our gazed-based bit allocation for networked video streaming. We use $QP = 10$ for a desired reference quality. For our gaze-based scheme (`hmm`) described in Section VI, the average QP outside the ROI is 15, as given by equation (23) and (24), where ϱ = 5e-09. An original scheme (`orig`) assigns $QP = 10$ for all blocks in a frame. The compressed frame size for the two schemes are given in Fig. 13 for CIF sequence `table` (`orig` bit-rate is 300kbps) and SD sequence `captain` (`orig` bit-rate is 800kbps). We see that in frames where the estimated state was tracking state `T`, fewer bits were allocated to non-ROI regions, resulting in bit-rate saving. In particular, we found that `hmm` achieved 20% and 29% bit saving compared to `orig` for sequence `table` and `captain`, respectively.

TABLE V

Comparing the saliency-based method with the HQ without real-time gaze tracking.

	FQ : *HQ*		
	Quiet-video	Busy-video	sum
votes	7:16	10:13	17:29
p-value	0.0461	0.5372	0.0698

F. Results for subjective testing

Of course, the bit saving must be achieved without significant loss of perceptual quality. To quantify this, we developed a real-time video coding / streaming system for subjective testing, with delay introduced between encoder and decoder to emulate $RTT = 50ms, 100ms, 150ms, 200ms, 250ms$. A Two Alternative Forced Choice (2AFC) method [50] was used to compare subjective video quality.

We first establish through subjective testing that using ROI-based video encoding without real-time gaze tracking / prediction will often not lead to sufficient perceptual quality. We performed the testing as follows. `FQ` encodes saliency objects in a video frame in high quality and other regions in low-quality, saving bit-rate. No gaze tracking / prediction is employed. `HQ` encodes entire video frames as the same high quality, resulting in a higher bit-rate. The subjective result could be seen in Table V.

We see in Table V that a substantially larger proportion of test subjects preferred `HQ` over `FQ`. That means test subjects were able to construe a difference in perceived visual quality between `HQ` and `FQ`. Looking more closely, this perceived difference in visual quality is most pronounced when the video content itself is quiet—steady state probability π_T is large.

We can explain the results as follows. It is clear that a pre-encoded ROI video coding scheme can handle gaze behavior of the mean user at best; idiosyncratic gaze behavior by individual users that deviate from the mean user—which happens more often for quiet videos—cannot be handled by offline encoded scheme. In contrast, our real-time gaze-based scheme can fully account for such personal idiosyncrasies, which explains our better subjective experimental results.

Next, to validate our gaze prediction strategy, two videos are randomly selected among the following three: the original HQ scheme `hq`, our proposed gaze-based ROI bit allocation scheme `hmm`, and the naïve linear prediction `nlp`. In each trial, participants looked at two videos back-to-back (with 3 seconds break in-between). Each video lasted for 10 seconds as recommended by ITU-R BT.500 [51]. After these presentations, each participant was asked to indicate which of the two videos looks better (First or Second), regardless of how certain they were of their response. Participants did not know which video was obtained by which kind of method. Full random combinations of two from `hq, hmm, nlp`, using 5 different `RTT`, gave a total of $2 \times 3 \times 5 = 30$ pairs.

The experiment was run in a quiet room with 23 participants (17 males and 6 females, and of age between 21 and 40). All participants had normal or corrected to normal vision. The illumination in the room was in the 300-320 Lux range. Each participant was familiarized with the task before the start of the experiment via a short instruction.

TABLE VI

Comparing the proposed method with the HQ and NLP method based on the subjective results at 5 different RTTs.

RTT/ms		*HMM : HQ*	*HMM : NLP*	*HQ : NLP*
50		23:23	37:9	40:6
	p-value	1	2.65E-07	1.82E-13
100		24:22	37:9	42:4
	p-value	0.7703	2.65E-07	8.08E-23
150		20:26	39:7	43:3
	p-value	0.3774	8.25E-11	3.36E-32
200		18:28	41:5	42:4
	p-value	0.1352	3.35E-17	8.08E-23
250		13:33	41:5	44:2
	p-value	0.0012	3.35E-17	5.68E-51

During video playback, the viewer's gaze points were tracked and sent to the streaming server.

The subjective testing results are shown in Table VI, where we indicate the number of responses showing preference for `hq, hmm, nlp` at different ***RTT*** values. We used the two-sided chi-square χ^2 test [52] to examine the statistical significance of the results. The null hypothesis is that there is no preference for either two of `HQ, HMM, NLP`. Under this hypothesis, the expected number of votes is 23 for each method. The p-value [52] is also indicated in the table. In experimental sciences, as a rule of thumb, the null hypothesis is rejected when $p < 0.05$. When this happens in Table VI, it means that the two methods cannot be considered to have the same subjective quality, since one of them has obtained a statistically significantly higher number of votes, and therefore seems to have better quality.

As seen in Table VI, in all of the pairs of `HMM-NLP` and `HQ-NLP`, the p-value is much smaller than 0.05, which indicates that subjects showed a statistically significant preference for our proposed method `HMM` and `HQ`. Further, looking across all pairs of `HMM-HQ`, the results show that participants only noticed significant difference when RTT is larger than $200ms$.

Our results clearly shows that our proposed method is always superior to `nlp`. Furthermore, it can achieve about 29% bit savings compared to HQ with only minor loss of subjective quality.

VIII. Conclusion

To improve the efficacy of gaze-based networked systems, in this paper, we proposed a hidden Markov model (HMM)-based gaze prediction strategy to predict future gaze locations one round-trip-time (RTT) into the future. The two HMM states correspond to two of human's intrinsic gaze behavioral movements. HMM parameters are derived offline by analyzing the video's visual saliency maps of per-pixel visual attention weights. The most likely HMM state is estimated via the forward algorithm (FA) using real-time collected gaze data. Given an estimated state, a prediction strategy using Kalman filtering is used to predict future gaze location. To validate our gaze prediction strategy, we apply our model to the bit allocation problem for network video streaming based on region of interest (ROI). Experiments show that bit rate can be reduced by up to 29% without noticeable visual quality degradation for RTT as high as 200ms.

For future work, we are investigating the joint tracking and prediction of human eye gaze and head position. We conjecture that the two movements—taken by the same human video observer—are correlated, and hence jointly tracking and predicting both can lead to better overall performance than optimizing them individually.

REFERENCES

[1] Y. Sugano, Y. Matsushita, Y. Sato, and H. Koike, “An incremental learning method for unconstrained gaze estimation,” in *European Conference on Computer Vision (ECCV2008)*, October 2008, pp. 656–667.

[2] M. Reale, T. Hung, and L. Yin, “Pointing with the eyes: Gaze estimation using a static/active camera system and 3D iris disk model,” in *IEEE International Conference on Multimedia and Expo*, Singapore, July 2010.

[3] LC Technologies, Inc., “Eyegaze Systems,” Fairfax, VA, http://www.eyegaze.com.

[4] Tobii Technology AB, “Eye tacking and eye control for research, communication and integration,” Danderyd, Sweden, http://www.tobii.com.

[5] A. Sippl, C. Holzmann, D. Zachhuber, and A. Ferscha, “Real-time gaze tracking for public displays,” in *Lecture Notes in Computer Science*, vol. 6439/2010, 2010, pp. 167–176.

[6] M. Reale, P. Liu, and Y. Lijun, “Using eye gaze, head pose, and facial expression for personalized non-player character interaction,” in *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, Colorado Springs, CO, June 2011.

[7] L. Loschky and G. Wolverton, “How late can you update gaze-contingent multiresolution displays without detection?” in *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 3, no.7, December 2007.

[8] A. Duchowski and A. Coltekin, “Foveated gaze-contingent displays for peripheral LOD management, 3D visualization, and stereo imaging,” in *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 3, no.4, December 2007.

[9] A. Duchowski, *Eye Tracking Methodology: Theory and Practice*. Springer, 2007.

[10] L. Itti, C. Koch, and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no.11, November 1998, pp. 1254–1259.

[11] L. Itti, “Automatic foveation for video compression using a neurobiological model of visual attention,” in *IEEE Transactions on Image Processing*, vol. 13, no.10, October 2004, pp. 1304–1318.

[12] O. L. Meur, P. L. Callet, and D. Barba, “Predicting visual fixations on video based on low-level visual features,” in *Vision Research*, vol. 47, no.19, September 2007, pp. 2483–2498.

[13] R. Faragher, “Understanding the basis of the Kalman filter via a simple and intuitive derivation,” in *IEEE Signal Processing Magazine*, vol. 29, no.5, September 2012, pp. 128–132.

[14] S. Floyd and K. Fall, “Promoting the use of end-to-end congestion control in the internet,” in *IEEE/ACM Trans. Networking*, August 1999.

[15] S. Floyd, M. Handley, J. Padhye, and J. Widmer, “Equation-based congestion control for unicast applications,” in *ACM SIGCOMM*, Stockholm, Sweden, August 2000.

[16] N. Jayant, J. Johnston, and R. Safranek, “Signal compression based on models of human perception,” in *Proceedings of the IEEE*, vol. 81, no.10, October 1993, pp. 1385–1422.

[17] W. Geisler and J. Perry, “A real-time foveated multiresolution system for low-bandwidth video communication,” in *SPIE Proceedings*, vol. 3299, July 1998.

[18] Y. Liu, Z. G. Li, and Y. C. Soh, “Region-of-interest based resource allocation for conversational video communication of H.264/AVC,” in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no.1, January 2008, pp. 134–139.

[19] Z. Chen and C. Guillemot, “Perceptually-friendly H.264/AVC video coding,” in *IEEE International Conference on Image Processing*, Cairo, Egypt, November 2009.

[20] “Opengazer: open-source gaze tracker for ordinary webcams,” http://www.inference.phy.cam.ac.uk/opengazer/.

[21] Y. Sugano, Y. Matsushita, and Y. Sato, “Calibration-free gaze sensing using saliency maps,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, San Francisco, June 2010.

[22] J. Chen and Q. Ji, “Probabilistic gaze estimation without active personal calibration,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, Troy, NY, June 2011.

[23] O. Komogortsev and J. Khan, “Perceptual multimedia compression based on predictive Kalman filter eye movement modeling,” in *ACM Multimedia Computing and Networking Conference*, San Jose, CA, January 2007.

[24] ——, “Eye movement prediction by Kalman filter with integrated linear horizontal oculomotor plant mechanical model,” in *Eye Tracking Research & Applications Symposium*, Savannah, GA, March 2008.

[25] O. V. Komogortsev and J. Khan, “Eye movement prediction by oculomotor plant Kalman filter with brainstem control,” in *Journal of Control Theory and Applications*, vol. 7, no.1, January 2009.

[26] R. Peters and L. Itti, “Computational mechanism for gaze direction in interactive visual environments,” in *Proceedings of the Symposium on Eye Tracking Research & Applications*, San Diego, CA, March 2006.

[27] M. C. et al., “Predicting human gaze using low-level saliency combined with face detection,” in *Neural Information Processing Systems*, 2007.

[28] O. L. Meur, P. L. Callet, D. Barba, and D. Thoreau, “A coherent computational approach to model the bottom-up visual attention,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no.5, May 2006, pp. 802–817.

[29] A. Bur and H. Hugli, “Optimal cue combination for saliency computation: A comparison with human vision,” in *Lecture Notes in Computer Science*, vol. 4528. Springer Verlag, 2007, pp. 109–118.

[30] A. Oliva, A. Torralba, M. S. Castelhano, and J. M. Henderson, “Top-down control of visual attention in object detection,” in *IEEE International Conference on Image Processing*, Barcelona, Spain, September 2003.

[31] N. Bruce and J. Tsotsos, “Saliency, attention, and visual search: An information theoretic approach,” in *Journal of Vision*, vol. 9, no.3, March 2009, pp. 1–24.

[32] D. Gao, V. Mahadevan, and N. Vasconcelos, “On the plausibility of the discriminant center-surround hypothesis for visual saliency,” in *Journal of Vision*, vol. 8, no.7, March 2008, pp. 1–18.

[33] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, “SUN: A bayesian framework for saliency using natural statistics,” in *Journal of Vision*, vol. 8, no.7, December 2008, pp. 1–20.

[34] L. Itti and P. Baldi, “Bayesian surprise attracts human attention,” in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2006, pp. 802–817.

[35] O. L. Meur and P. L. Callet, “What we see is most likely to be what matters: Visual attention and applications,” in *IEEE International Conference on Image Processing*, Cairo, Egypt, November 2009.

[36] R. Valenti, N. Sebe, and T. Gevers, “What are you looking at? improving visual gaze estimation by saliency,” in *International Journal on Computer Vision*, vol. DOI 10.1007/s11263-011-0511-6, 2011.

[37] “iLab neuromorphic vision C++ toolkit,” http://ilab.usc.edu/toolkit/downloads.shtml.

[38] D. K. N. Doulamis, A. Doulamis and S. Kollias, “Low bit-rate coding of image sequences using adaptive regions of interest,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 8, pp. 928–34, 1998.

[39] N. Bruce and P. Kornprobst, “On the role of context in probabilistic models of visual saliency,” in *IEEE International Conference on Image Processing*, Cairo, Egypt, November 2009.

[40] S. Davies, D. Agrafiotis, C. Canagarajah, and D. Bull, “A gaze prediction technique for open signed video content using a track before detect algorithm,” in *IEEE International Conference on Image Processing*, San Diego, CA, October 2008.

[41] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[42] R. J. Leigh and D. S. Zee, *The Neurology of Eye Movements*. Oxford University Press, 2006.

[43] D. Sun, S. Roth, and M. Black, “Secrets of optical flow estimation and their principles,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, June 2010.

[44] *Video Coding for Low Bitrate Communication*, ITU-T Recommendation H.263, February 1998.

[45] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no.7, July 2003, pp. 560–576.

[46] Y. Feng, G. Cheung, W. t. Tan, and Y. Ji, “Hidden Markov model for eye gaze prediction in networked video streaming,” in *IEEE International Conference on Multimedia and Expo*, Barcelona, Spain, July 2011.

[47] M. V. Venkatesh and S. c. S. Cheung, “Eye tracking based perceptual image inpainting quality analysis,” in *IEEE International Conference on Image Processing*, Hong Kong, September 2010.

[48] “IRCCyN lab platforms & databases,” http://www.irccyn.ec-nantes.fr/ lecallet/platforms.htm.

[49] Y. Feng, G. Cheung, P. L. Callet, and Y. Ji, “Video attention deviation estimation using inter-frame visual saliency map analysis,” in *IS&T/SPIE Visual Information Processing and Communication Conference*, Burlingame, CA, January 2012.

[50] M. Taylor and C. Creelman, “PEST: Efficient estimates on probability functions,” *J. Acoustical Society of America*, vol. 41, pp. 782–787, 1967.

[51] ITU-R, “Recommendation BT.500-8: Methodology for the subjective assessment of the quality of television pictures,” ITU, Tech. Rep., 1998.

[52] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, 2007.

Yunlong Feng received B.S. degree in computer science from University of Science and Technology of China (USTC) in 2009. Currently, he is a Ph.D Candidate in Department of Informatics at the Graduate University for Advanced Studies (SOKENDAI), Tokyo Japan.

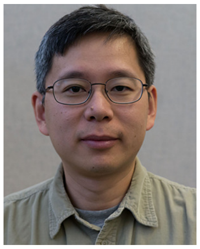
He is working as a Research Assistant at National Institute of Informatics (NII) in Tokyo Japan, since 2009. His research interests include interactive system for networked video streaming, in particular, gaze analysis and prediction, saliency map analysis etc.

He is a author of the best paper finalists in ICME 2011. He received the ICM English Session Encouragement Award in 2012, and SOKENDAI President’s Award in 2013. He has been a visiting scholar of Simon Fraser University (SFU), Canada with the host of Prof. Ivan V. Bajic in 2012, and IRCCyN at Nante France with the host of Prof. Patrick Le Callet in 2013 respectively. He has also be supported by SOKENDAI short-time study abroad programm to the Mobile and Immersive Experience Lab at Hewlett Packard Laboratory with the supervision of Dr. Wai-tian Tan at Palo Alto, USA in 2012.

Gene Cheung (M’00—SM’07) received the B.S. degree in electrical engineering from Cornell University in 1995, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of California, Berkeley, in 1998 and 2000, respectively.

He was a senior researcher in Hewlett-Packard Laboratories Japan, Tokyo, from 2000 till 2009. He is now an associate professor in National Institute of Informatics in Tokyo, Japan.

His research interests include 3D visual representation, single- / multiple-view video coding & streaming, and immersive communication. He has served as associate editor for IEEE Transactions on Multimedia from 2007 to 2011 and currently serves as associate editor for DSP Applications Column in IEEE Signal Processing Magazine and APSIPA journal on signal & information processing, and as area editor for EURASIP Signal Processing: Image Communication. He currently serves as member of the Multimedia Signal Processing Technical Committee (MMSP-TC) in IEEE Signal Processing Society (2012-2014). He has also served as area chair in IEEE International Conference on Image Processing (ICIP) 2010, 2012-2013, technical program co-chair of International Packet Video Workshop (PV) 2010, track co-chair for Multimedia Signal Processing track in IEEE International Conference on Multimedia and Expo (ICME) 2011, symposium co-chair for CSSMA Symposium in IEEE GLOBECOM 2012, and area chair for ICME 2013. He is a co-author of best student paper award in IEEE Workshop on Streaming and Media Communications 2011 (in conjunction with ICME 2011), best paper finalists in ICME 2011 and ICIP 2011, and best paper runner-up award in ICME 2012.



Wai-tian Tan received the B.S. degree in electrical engineering from Brown University, in 1992, the M.S.E.E. degree from Stanford University in 1993, and the Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, in 2000. He has been with Hewlett Packard Laboratories, Palo Alto since 2000, and is currently a principal research scientist. His research interests are in developing adaptive streaming systems under different practical constraints, and more generally in coding, communications, and new applications of video. He has over 50 research papers, holds 14 patents, and has developed production multimedia software for mobile devices and telecommunication networks.

Patrick Le Callet received M.Sc. degree PhD degree in image processing from Ecole polytechnique de luniversit de Nantes. He was also student at the Ecole Normale Superieure de Cachan where he get the Aggrgation (credentialing exam) in electronics of the French National Education. Since 2003 is teaching at Ecole polytechnique de luniversit de Nantes in the Electrical Engineering and the Computer Science department where is now Full Professor. Since 2006, he is the head of the Image and VideoCommunication lab at CNRS IRCCyN, a group of more than 35 researchers. He is mostly engaged in research dealing with the application of human vision modeling in image and video processing. His current centers of interest are 3D image and video quality assessment, watermarking techniques and visual attention modeling and applications. He is co-author of more than 190 publications and communications and co-inventor of 13 international patents. He has coordinated and is currently managing for IRCCyN several National or European collaborative research programs. He is serving in VQEG (Video Quality Expert Group) where is co-chairing the ”HDR Group” and ”3DTV” activities. He is currently serving as associate editor for IEEE transactions on Circuit System and Video Technology, SPIE Journal of Electronic Imaging and SPRINGER EURASIP Journal on Image and Video Processing.

Yusheng Ji received B.E., M.E., and D.E. degrees in electrical engineering from the University of Tokyo. She joined the National Center for Science Information Systems, Japan (NACSIS) in 1990. Currently, she is a Professor at the National Institute of Informatics, Japan (NII), and the Graduate University for Advanced Studies (SOKENDAI). Her research interests include network architecture, traffic control, and performance analysis for quality of service provisioning in wired and wireless communication networks. She is a member of IEEE, IEICE, and IPSJ. She has served as a Board Member of Trustees of IEICE, Steering Committee Member of Quality Aware Internet (QAI) SIG and Internet and Operation Technologies (IOT) SIG of IPSJ, Associate Editor of IEICE Transactions, Guest Editor and Guest Associate Editor of Special Sections of IEICE Transactions, Associate Editor of IPSJ Journal, Guest Associate Editor of Special Issues of IPSJ Journal, etc. She has also served as a TPC member of many conferences, including IEEE ICC, GLOBECOM, PIMRC, WCNC, VTC etc., and the Wireless Networking Symposium Co-chair of IEEE GLOBECOM 2012. She is currently serving as an Editor of IEEE Transactions of Vehicular Technology, an Expert Member of IEICE Technical Committees on Internet Architecture, and Communication Quality.