Video Attention Deviation Estimation using Inter-Frame Visual Saliency Map Analysis

Yunlong Feng[a], Gene Cheung[b], Patrick Le Callet[c], Yusheng Ji[b]

[a]The Graduate University for Advanced Studies;
[b]National Institute of Informatics;
[c]Université de Nantes

ABSTRACT

A viewer's visual attention during video playback is the matching of his eye gaze movement to the changing video content over time. If the gaze movement matches the video content (e.g., follow a rolling soccer ball), then the viewer keeps his visual attention. If the gaze location moves from one video object to another, then the viewer shifts his visual attention. A video that causes a viewer to shift his attention often is a "busy" video. Determination of which video content is busy is an important practical problem; a busy video is difficult for encoder to deploy region of interest (ROI)-based bit allocation, and hard for content provider to insert additional overlays like advertisements, making the video even busier. One way to determine the busyness of video content is to conduct eye gaze experiments with a sizable group of test subjects, but this is time-consuming and cost-ineffective. In this paper, we propose an alternative method to determine the busyness of video—formally called video attention deviation (VAD): analyze the spatial visual saliency maps of the video frames across time. We first derive transition probabilities of a Markov model for eye gaze using saliency maps of a number of consecutive frames. We then compute steady state probability of the saccade state in the model—our estimate of VAD. We demonstrate that the computed steady state probability for saccade using saliency map analysis matches that computed using actual gaze traces for a range of videos with different degrees of busyness. Further, our analysis can also be used to segment video into shorter clips of different degrees of busyness by computing the Kullback-Leibler divergence using consecutive motion compensated saliency maps.

Keywords: Visual attention, eye gaze tracking, visual saliency

1. INTRODUCTION

During playback of a video clip displayed on a reasonably large screen, a viewer sitting at a comfortably close distance from the screen cannot observe all spatial regions simultaneously and clearly in a given video frame. In fact, it has been shown[1] that the ability of a viewer to discern details away from his focal point of visual attention drops off precipitously as a function of the viewing angle. So driven by top-down motivation (e.g., a task in mind) and/or bottom-up stimulus (e.g., low-level features of the visual scene), a viewer typically *shifts* his visual focal attention from time to time, to study new local spatial regions of interest. This movement is known as *saccade* in eye gaze literature.[2] Saccade contrasts with another eye gaze movement *pursuit*, where a viewer's gaze point simply follows a moving video object, like a rolling soccer ball across the screen. In the latter case, the viewer does not shift his visual attention, but rather, maintains his attention on the same video object.

Thus, one can determine the extent in which a viewer shifts his visual attention by comparing his eye gaze trajectory with the video content being observed.[3] It is apparent that different video contents contain different degrees of bottom-up stimulus, inducing different amount of visual attention shifts from viewers. A video content that induces very few shifts of visual attention from the viewer, like a stationary camera, head-and-shoulders presidential address, is called a "quiet" video. On the contrary, a video content that induces frequent shifts of visual attention, like a dance music video, is called a "busy" video.

Further author information: (Send correspondence to Yunlong Feng, Gene Cheung)
Yunlong Feng: E-mail: fengyl@nii.ac.jp, Telephone: +81 (3) 4212 2688
Gene Cheung: E-mail: cheung@nii.ac.jp, Telephone: +81 (3) 4212 2652

Determination of which video contents are busy is an important practical problem. For example, in a region of interest (ROI)-based bit allocation scheme,[3–5] more bits are allocated to the spatial region containing the viewer's current focal attention point, and fewer bits elsewhere, so that the overall bitrate can be reduced *without* degrading perceptual visual quality. If a video is busy, frequent saccade movements—which are known to be very fast and unpredictable in speed and duration[6]—will limit the effectiveness of such ROI-based allocation scheme.[3] Identified busy videos can then be encoded in more traditional methods to reduce bitrate, e.g., using uniform quantization across the entire frame.

In another example, given an identified busy video, a content provider is limited by the extent in which visual overlays[7] like advertisements can be painted on top. Doing so will make the video even busier, as too many objects compete for viewer's visual attention. For an identified busy video content, advertisements can instead be inserted as full frames temporally, extending the running time of the video.

One straightforward way to determine the busyness of a video content is to conduct eye gaze experiments, using a real-time gaze tracking system,[8] with a sizable group of test subjects. This, however, is clearly too time-consuming and cost-ineffective for a large number of video contents. In this paper, we propose an alternative method to determine the busyness of video—formally called *video attention deviation* (VAD)—by analyzing the visual saliency maps[9] of individual video frames across time. Saliency maps, grey-scale images that reflect per-pixel visual attention weight in original video frames, are constructed using combinations of low-level neuronal feature maps—e.g., color and intensity contrasts—that have been found to attract attention in humans and monkeys. Saliency maps that are computed for individual video frames describe visual attention deviation *spatially*. In contrast, we seek to describe visual attention deviation of a video *temporally*, i.e., how often a viewer will likely shift attention over time.

Our methodology is as follows. First, we derive transition probabilities of a Markov model using saliency maps of a number of consecutive frames. Then, we compute steady state probability of the saccade state in the model, which becomes our estimate of VAD. We demonstrate that the computed steady saccade state using saliency map analysis matches that computed from actual gaze traces for a range of videos with different degrees of busyness. Further, our inter-frame saliency map analysis can also be used to segment a piece of video content into shorter clips of different degrees of busyness by computing the Kullback-Leibler (KL) divergence using consecutive motion compensated saliency maps.

We outline the paper as follows. We first overview related work in Section 2. We then discuss our previously proposed Markov model for eye gaze movement in Section 3. We discuss our derivation of saccade steady state probability in the Markov model using computed saliency maps in Section 4. Finally, experimental results and conclusion are presented in Section 5 and 6, respectively.

2. RELATED WORK

Visual attention (VA) modeling is the focus of many research efforts in the last decade, following attempts in the vision science & perception community, to better understand the fundamentals of visual attention. Several computational models[9, 10] to emulate VA have been consequently proposed which detect the locations that attract the eye gaze. Most of the models compute a saliency map that value each pixel according to its visual saliency.

More recent computation of saliency maps for video frames also considers low-level temporal features such as motion and flicker.[10] However, our goal here is not to propose new models of visual saliency maps, but to use saliency maps, computed using previously established techniques, to derive VAD offline in a computationally efficient way. In this paper, we compute saliency maps using methodology in Itti et al.,[9] based on a plausible model of bottom-up visual attention. This model offers good performance with reasonable computational cost. An existing implementation of the model is available online.[11]

3. HIDDEN MARKOV MODEL FOR EYE GAZE

In this section, we briefly discuss our previously proposed hidden Markov model (HMM)[12] for eye gaze movement during a viewer's observation of a video.[3] The HMM has three latent states: fixation, pursuit, and saccade. State F (*fixation*) models the case when eye gaze is fixated at a stationary object in the video. State P (*pursuit*) models
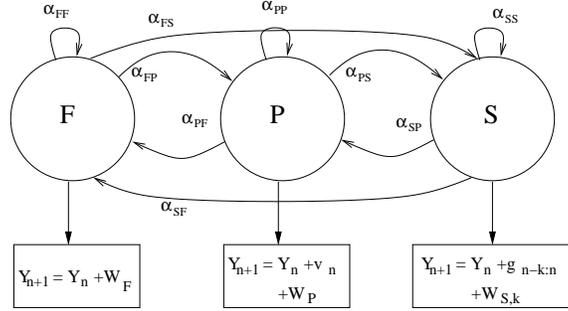
Figure 1. Hidden Markov model for eye gaze during video observation. Circles denote latent states of `F` (fixation), `P` (pursuit) and `S` (saccade). Boxes denote observations.

the case when gaze follows motion of a moving object. State `S` (*saccade*) models the rapid transition from one fixation point to another. These are the three major types of eye movements for the human eye.[2]

Figure 2. Eye gaze data on frame 220 of MPEG test sequence `kids`. Eye gaze data is marked by a white 5×5 square.

An HMM is Markovian in that the determination of state variable X_{n+1} at time $n + 1$ (`F`, `P` or `S`) 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 (gaze locations) are observed, where each Y_n is generated by a random process dependent on current latent state $X_n = i$. Random process for gaze tracking means modeling noise in the gaze data; since instability of human vision and inaccuracy of gaze tracking algorithms mean non-negligible noise is present in the eye gaze data. As an example, see Fig. 2 where a viewer is looking at the red ball, but the gaze tracker returns a gaze data point slightly away from the ball.

For this paper, we are interested only in the derivation of the state transition probabilities $\alpha_{i,j}$'s in the HMM through analysis of the visual saliency maps. We discuss this next.
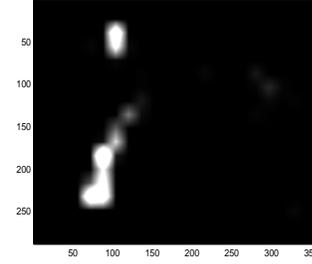
4. ANALYSIS OF VISUAL SALIENCY MAPS

4.1 Finding Saliency Objects

We first compute visual saliency maps for all video frames using methodology in Itti et al.[9] As an example, in Fig. 3 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.

Having computed visual saliency maps, we first normalize each one, so the sum of all saliency values in a frame 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-block 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 in the given frame. A viewer may of course have gaze location 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
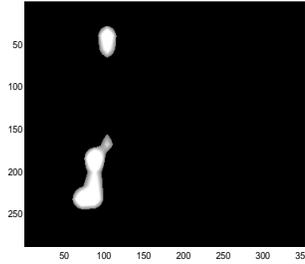
(a) original video frame

(b) corresponding saliency map

Figure 3. Original video frame 157 of MPEG test sequence `table`, and the corresponding visual saliency map, calculated using method in Itti et al.[9]

(a) saliency map w/ threshold

(b) saliency objects

Figure 4. Normalized saliency map after applying threshold, and resulting salient objects in video frame 157 of sequence `table`.

time. Returning to our earlier example, we see in Fig. 4(a) the normalized saliency map with normalized saliency values below threshold τ_s set to zero. Correspondingly, we see the saliency objects in Fig. 4(b).
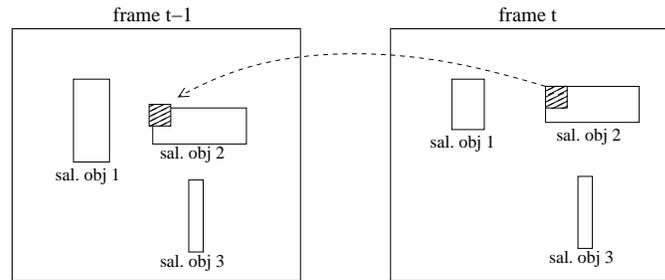

Figure 5. Example of how correspondence of saliency objects located in pairs of consecutive saliency maps are found using ME.

We can establish correspondence among saliency objects in consecutive frames using motion estimation (ME), commonly used in video compression algorithms like H.26x. 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 best match 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 of $o_{t,i}$ map to blocks of the same object $o_{t-1,j}$, then we declare they are the same object. If no such object exists in previous map $t-1$, then we declare object $o_{t,i}$ to be a new object appearing for the first time in map t. As an example, in Fig. 5, we see that a block in object 2 in frame t has found a matching block in object 2 in frame $t-1$.

Having established correspondence among saliency objects in consecutive frames, we can also use the same motion information to label each one either as a stationary or moving object based on the mean motion vector magnitude of matched blocks. A viewer's gaze following a saliency object that is stationary or moving will be in state fixation or pursuit, respectively.

4.2 Deriving Transition Probabilities

Having identified saliency objects across frames, we now derive state transition probabilities in the eye gaze HMM in this section. For simplicity, we will assume state transition probabilities of F to P and P to F, α_{FP} and α_{PF}, are both zero. These represent cases when a viewer switches from one saliency object to another faster than the video frame rate. We have found in practice that while these cases do happen, they happen rarely.[3]

For intuition, we first describe the derivation of state transition probabilities for a particular case. Suppose there are three saliency objects in consecutive frames t and $t+1$, with probabilities $p_{t,1}$, $p_{t,2}$ and $p_{t,3}$ respectively in frame t, and probabilities $p_{t+1,1}$, $p_{t+1,2}$ and $p_{t+1,3}$ respectively in 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 object(s)), we can calculate the relative probability size of objects by comparing their volumes in each frame:

$$
\begin{aligned}
s_t &= \beta_{t,1}\ p_{t,1} = \beta_{t,2}\ p_{t,2} = \beta_{t,3}\ p_{t,3} \\
s_{t+1} &= \beta_{t+1,1}\ p_{t+1,1} = \beta_{t+1,2}\ p_{t+1,2} = \beta_{t+1,3}\ p_{t+1,3}
\end{aligned}
\tag{1}
$$

where β's are the scaling factors among objects in each frame. See Fig. 4(a) again for example of volume of visual saliency objects.

Further, we know that the law of total probability dictates that the sum of probabilities in each frame must equal to 1:

$$
\begin{aligned}
p_{t,1} + p_{t,2} + p_{t,3} + s_t &= 1 \\
p_{t+1,1} + p_{t+1,2} + p_{t+1,3} + s_{t+1} &= 1
\end{aligned}
\tag{2}
$$

Together with earlier derived (1), we can determine the 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. Suppose that we have determined through ME that object 1 is a stationary object, while object 2 and object 3 are moving objects. 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:

$$
p_{t+1,1} = p_{t,1}\ \alpha_{FF} + s_t\ \alpha_{SF}
\tag{3}
$$

We can write similar equations for probabilities $p_{t+1,2}$ and $p_{t+1,3}$ of moving objects 2 and 3 in frame $t+1$. The only complication here is that the probability $s_t\ \alpha_{SP}$ from state S to P must be split between object 2 and 3, according to their relative volumes:

$$
p_{t+1,2} = p_{t,2}\ \alpha_{PP} + s_t\ \alpha_{SP}\ \left(\frac{\beta_{t+1,2}}{\beta_{t+1,2} + \beta_{t+1,3}} \right)
\tag{4}
$$

$$
p_{t+1,3} = p_{t,3}\ \alpha_{PP} + s_t\ \alpha_{SP}\ \left(\frac{\beta_{t+1,3}}{\beta_{t+1,2} + \beta_{t+1,3}} \right)
\tag{5}
$$

We can similarly write state transition equation for the saccade state as well:

$$
s_{t+1} = p_{t,1}\ \alpha_{FS} + p_{t,2}\ \alpha_{PS} + p_{t,3}\ \alpha_{PS} + s_t\ \alpha_{SS}
\tag{6}
$$

Note that we have now four state transition equations for the seven 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:

$$
\begin{aligned}
\alpha_{FF} + \alpha_{FS} &= 1 \\
\alpha_{PP} + \alpha_{PS} &= 1 \\
\alpha_{SS} + \alpha_{SF} + \alpha_{SP} &= 1
\end{aligned}
\tag{7}
$$

These three linear equations, together with the earlier derived four linear state transition equations, means the seven state transition probabilities α's can be computed by solving a set of linear equations.

More generally, when there are fewer than three saliency objects in two consecutive frames, we have fewer linear equations than variables (α's). In that case, we will use more consecutive frames to construct enough linear equations. If there are more than three saliency objects, then we have more linear equations than variables. In that case, we 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:

$$
C\mathbf{a} = \mathbf{b} + \mathbf{n}
\tag{8}
$$

where $\mathbf{a} = [\alpha_{FF}, \alpha_{FS}, \ldots]^T$ 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:

$$
\mathbf{a}^* = C^+\mathbf{b}
\tag{9}
$$

where $C^+ = (C^T C)^{-1} C^T$ is the Moore-Penrose pseudo-inverse of matrix C. Having computed the transition probabilities α's, for a consecutive number of saliency maps starting from map t, we can compute the steady state probabilities π_t of the HMM by performing eigen-analysis as typically done in the literature. VAD is the average steady state probability for saccade π_S over all the frames in the video clip.

4.3 Computing Kullback-Leibler Divergence

In addition, we can track how fast viewer's gaze busyness is changing by computing the Kullback-Leibler (KL) divergence using consecutive saliency maps of the video content. Using computed KL divergence, we can optionally divide a video into segments of more stationary gaze statistics, each with its own VAD.

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 for the pursuit state.

We compute KL divergence by treating saliency map π_t at t and motion-compensated saliency map π_{t+1} at $t+1$ as probability distribution functions. Mathematically, we compute the KL divergence as follows:

$$
d_{KL}(\pi_t || \pi_{t+1}) = \sum_i \pi_t(i) \log\left(\frac{\pi_t(i)}{\pi_{t+1}(i)}\right)
\tag{10}
$$

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

5. EXPERIMENTATION

To show the potential of our proposed VAD estimation using inter-frame visual saliency map analysis, we used four test sequences as input to USC's visual saliency map calculation software:[11] i) two 300-frame standard MPEG video test sequences, `silent` and `table`, at CIF resolution (352×288) and ii) two 250-frame higher resolution video sequences, `captain` and `group`, at SD resolution (720×576) and downloadable from IRCCyN Lab website.[13] All videos have 30 frames per second (fps) playback speed. `silent` and `captain` are "quiet" videos with few visual activities, while `table` and `group` are "busy" videos with lots of visual activities. Single frames of `captain` and `group` are shown in Fig. 6.

(a) frame 15 of SD test sequence `captain`   (b) frame 199 of SD test sequence `group`

Figure 6. Sample frame of SD resolution sequence
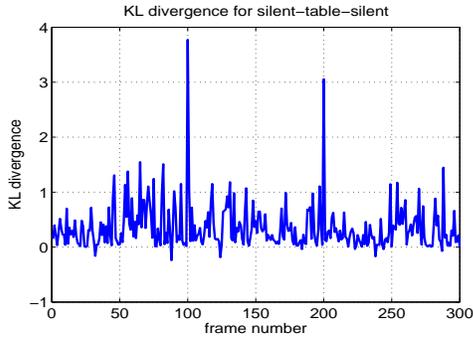Table 1. Comparison of computed VAD values

	π_S
gaze data for `Silent` (300 frames)	0.063
saliency map analysis for `Silent` (300 frames)	0.089
gaze data for `Table` (300 frames)	0.432
saliency map analysis for `Table` (300 frames)	0.442
gaze data for `Captain` (250 frames)	0.152
saliency map analysis for `Captain` (250 frames)	0.181
gaze data for `Group` (250 frames)	0.439
saliency map analysis for `Group` (250 frames)	0.457

We first validate our proposed saliency map analysis discussed in Section 4, i.e., whether saccade steady state probability (VAD) derived from saliency map analysis are roughly the same as ones obtained using actual eye-gaze data traces. To obtain ground truth gaze data, a trained user performed multiple viewings of each test sequence, each time continuously recorded his intention of fixation, pursuit or saccade by pressing keys on a keyboard. Using this "ground truth" data, we calculated one set of state transition probabilities in the HMM and then the saccade steady state probability π_s. The saccade steady state probabilities for all test sequences are shown in Table 1. Notice that `silent` and `captain` are indeed a quieter video: the saccade steady state probability π_S is much smaller than `table` and `group`.
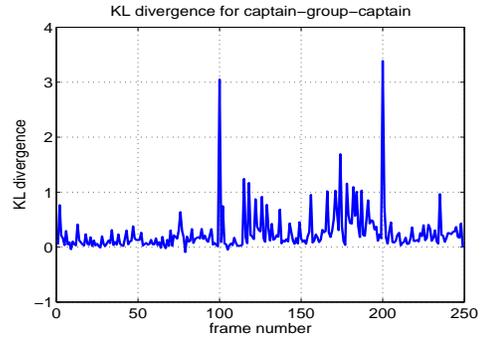
For comparison, we see that the derived HMM parameters using saliency maps analysis are quite close to the ground truth gaze data trace. In particular, we see that the analytical saccade steady state probability π_S for both `silent` and `table` are very close to the ground truth trace numbers, even though π_S's for `silent` and `table` are very different. This shows accuracy of our proposed saliency map analysis. We also performed the same experiment for the two SD test sequences. We again see very similar numbers between saccade steady state probabilities derived using saliency map analysis and ones obtained using eye-gaze data trace.

We now verify our claim that video can be partitioned into segments of roughly stationary gaze statistics using computed KL divergence of motion-compensated saliency maps, as discussed in Section 4.3. For testing, we constructed a composite CIF video clip consisting of 100-frame of `silent`, plus 100-frame of `table`, plus 100-frame of `silent`. Because 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 frames 101 and 201. Similarly, we constructed a composite SD video clip consisting of 100-frame of `captain`, plus 100-frame of `group`, plus 50-frame of `captain`.

In Fig. 7(a) and (b), we see the computed KL divergence as function of frame number for the two composite video clips. We clearly see the spikes around frame 101 and 201, indicating a change in gaze statistics. This validates our claim that KL divergence using motion-compensated saliency maps can be used to partition video into segments of roughly stationary gaze statistics.

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

Figure 7. KL Divergence as function of frame numbers

6. CONCLUSION

A "busy" video is a video that induces a viewer to shift his visual attention often during playback. Identification of busy video, without conducting expensive and cumbersome eye-gaze subjective testings, is a technical challenge. In this paper, we propose a method to determine the busyness of video—called video attention deviation (VAD)— by analyzing the visual saliency maps of video frames across time. In our experiments, we demonstrated that the computed VAD matches that computed using actual gaze traces for a range of videos with different degrees of busyness, demonstrating the effectiveness of our proposed scheme.

REFERENCES

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

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

[3] Feng, Y., Cheung, G., t. Tan, W., and Ji, Y., "Hidden Markov model for eye gaze prediction in networked video streaming," in [*IEEE International Conference on Multimedia and Expo*], (July 2011).

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

[5] Chen, Z. and Guillemot, C., "Perceptually-friendly H.264/AVC video coding," in [*IEEE International Conference on Image Processing*], (November 2009).

[6] "Saccade." http://en.wikipedia.org/wiki/Saccade.

[7] Cheung, G., Tan, W.-T., Shen, B., and Ortega, A., "ECHO: A community video streaming system with interactive visual overlays," in [*IS&T/SPIE 15th Aunnual Multimedia Computing and Networking (MMCN'08)*], (January 2008).

[8] "Opengazer: open-source gaze tracker for ordinary webcams." http://www.inference.phy.cam.ac.uk/opengazer/.

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

[10] Meur, O. L., Callet, P. L., and Barba, D., "Predicting visual fixations on video based on low-level visual features," in [*Vision Research*], **47, no.19**, 2483–2498 (September 2007).

[11] "iLab neuromorphic vision C++ toolkit." http://ilab.usc.edu/toolkit/downloads.shtml.

[12] Bishop, C., [*Pattern Recognition and Machine Learning*], Springer (2006).

[13] "IRCCyN lab platforms & databases." http://www.irccyn.ec-nantes.fr/ lecallet/platforms.htm.