Learning video saliency from human gaze using candidate selection

Dmitry Rudoy
Technion
Haifa, Israel
dmitry.rudoy@gmail.com

Dan B Goldman
Adobe Research
Seattle, WA
dgoldman@adobe.com

Eli Shechtman
Adobe Research
Seattle, WA
elishe@adobe.com

Lihi Zelnik-Manor
Technion
Haifa, Israel
lihi@ee.technion.ac.il

Abstract

During recent years remarkable progress has been made in visual saliency modeling. Our interest is in video saliency. Since videos are fundamentally different from still images, they are viewed differently by human observers. For example, the time each video frame is observed is a fraction of a second, while a still image can be viewed leisurely. Therefore, video saliency estimation methods should differ substantially from image saliency methods. In this paper we propose a novel method for video saliency estimation, which is inspired by the way people watch videos. We explicitly model the continuity of the video by predicting the saliency map of a given frame, conditioned on the map from the previous frame. Furthermore, accuracy and computation speed are improved by restricting the salient locations to a carefully selected candidate set. We validate our method using two gaze-tracked video datasets and show we outperform the state-of-the-art.

1. Introduction

Predicting where people look in video is relevant in many applications. For example, in advertising, it may be important for the producer to know if the key concept catches the viewer's eye [29]. Furthermore, if one knows where people are likely to look in a video, relevant content can be placed there. Another application that might take advantage of human gaze prediction is video editing [1]: knowing where the viewer looks could help to create smoother shot transitions. Moreover, we hypothesize that reliable gaze prediction may drive gaze-aware video compression or key-frame selection [15].

Image saliency is well explored in the computer vision community. It is known that color, high contrast and human subjects draw our attention [18]. When viewing an image over several seconds, a human observer can leisurely scan multiple areas of interest over time, and different viewers may observe various paths through the image content. In contrast, observers watching a video with dynamic content have only a fraction of a second to observe each frame. Hence they typically focus on the single most salient point of each frame [24]. The difference between human fixations when viewing a static image versus a video frame is exemplified in Figure 1. As can be seen, people watching the image for 3 seconds attended several faces, while people watching the same image as a frame within a video, focused on a single face, that of the speaker.

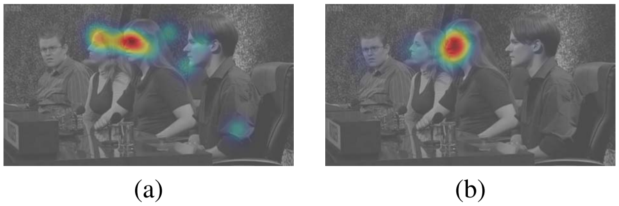


Figure 1. **Image vs. video saliency.** The same image was displayed to human observers twice: once static for 3 seconds (a), and once embedded within a video (b). The saliency maps overlayed on the images show that video saliency is tighter and more concentrated on a single object, while image saliency covers several interesting locations.

In this work we propose a method that predicts saliency by explicitly accounting for gaze transitions over time. Rather than trying to model where people look in each frame independently, we predict the gaze location given the previous frame's fixation map. In this way, we handle inter-frame dynamics of the gaze transitions, along with within-frame salient locations. To this end we learn a model that predicts a saliency map for a frame given the fixation map from a recent preceding moment and test it on a large set of realistic videos.

A key contribution of this work is the observation that saliency in video is typically very sparse and computing it at each and every pixel is redundant. Instead, we select a set of candidate gaze locations, and compute saliency only at these locations. The candidates are extracted using static, dynamic and semantic cues. We verify experimentally that our candidate-based approach outperforms the pixel based approach, and is significantly better than an image saliency based approach.

Another contribution of this work is an approach for learning conditional saliency. Since a video is a stream of frames, the human gaze in each frame depends on the previous gaze locations. This is different from images, where each image is assumed to be viewed independently. In this paper we suggest a method for learning the conditional probabilities across consecutive frames.

The rest of the paper is organized as follows. Section 2 reviews the previous work in psychology, image and video saliency. Section 3 provides a high-level overview of the proposed method. Section 4 explains candidate selection, and Section 5 focuses on learning the conditional probability. Experimental validation, together with comparison against the pixel-wise calculation, is presented in Section 6 and conclusions are drawn in Section 7.

2. Related work

Scientists have studied human visual attention for decades. In early works a separation between voluntary and involuntary attention is proposed [6]. Henderson [13] focus on understanding the image data, while Mital et al. [24] and Goldstein et al. [10] focus on videos. Others tried to understand the temporal effect of eye motion in videos. Some concentrate on the edit points, e.g., scene cuts, that have a large influence on the fixations in video [28]. Others try to understand the behavior within the shot and build a high-level theory [27].

Already in 1980, Treisman and Gelade [30] proposed a feature-integration theory, which aggregates several feature types. Later, Koch and Ullman [20] proposed a feed-forward model for the integration, along with the concept of a *saliency map* – a measure of visual attraction of every point in the scene. This idea was first implemented and verified by Itti et al. [16], who proposed one of the first complete models of human attention in images.

Since then much progress in image saliency has been made. Some of the models use only low-level information [12], while others use high level object detection [18] or context [9]. The reader is referred to a recent survey on the subject for more details [2].

Much less work has been done on video saliency. Gou et al. [11] adopt an efficient method based on spectral analysis of the frequencies in the video. Kim et al. [19] extend the center-surround approach for images to video by adding another dimension. A somewhat similar approach is proposed by Mahadevan and Vasconcelos [23] – they model video patches as dynamic textures to handle complicated backgrounds and moving camera. Seo and Milanfar [26] propose using self-resemblance in both static and space-time saliency detection. Cui et al. [8] take a different approach: they concentrate on motion saliency only and detect it by using temporal spectral analysis. Finally, Hou and Zhang [14] proposed using incremental coding length to measure the rarity of features. This method can find salient regions both in images and in videos.

Our work differs from previous video saliency methods by narrowing the focus to a small number of candidate gaze locations, and learning conditional gaze transitions over time.

3. Motivation and overview

Most previous saliency modeling methods calculate a saliency value for every pixel. In our work we propose to calculate saliency at a small set of candidate locations, instead of at every pixel. We motivate this based on two observations about patterns of human gaze.

First, we observe that image saliency studies concentrate on a single image stimulus, without any prior. This is usually achieved by “resetting” the participants’ gaze – presenting a black screen or a single target in the center. In video this is not a possible initial condition for real-world viewing. Here, the gaze varies little between frames, and when it does change significantly it is highly constrained to local regions. Therefore, our solution considers only a small number of plausible candidate regions, and treats the regions in aggregate, rather than per-pixel.

Our second observation is that when watching dynamic scenes people usually follow the action and the characters by shifting their gaze to a new interesting location in the scene. Focusing on a sparse candidate set of salient locations allows us to model and learn these transitions explicitly with a relatively small computational effort.

To accommodate these observations our system consists of three phases: identifying candidate gaze locations at each frame (Section 4), extracting features for those locations (Section 5.1) and learning or predicting gaze probabilities for each candidate (Section 5.3). Learning and inference follow the same three stages.

4. Candidate extraction

We start by presenting a method for detecting candidate regions. We consider three types of candidates. *Static candidates* indicate the locations that capture attention due to local contrast or uniqueness, irrespective of motion. *Motion candidates* reflect the areas that are attractive due to the motion between frames. Last, *semantic candidates* are those that arise from higher-level human visual processing.

The static and semantic candidate locations are generated separately for every video frame. The motion candidates are computed using optical flow between neighboring pairs of frames, and therefore implicitly account for the dynamics in the video. Each candidate location is represented by a Gaussian blob, characterized by the spatial coordinates of its mean and by its covariance matrix.

4.1. Static candidates

Since a video is composed of individual frames we start with candidates that attract peoples' attention due to static cues. For a given frame of interest we calculate the graph-based visual saliency (GBVS), proposed by Harel et al. [12]. We preferred GBVS over other image saliency methods for two main reasons: (i) it has been shown that GBVS accurately predicts human fixations in static images [3], and (ii) it is fast to calculate compared to more accurate methods [18]. We hypothesize that other image saliency detection methods may be used instead.

Given the image saliency map we wish to find the most attractive candidate regions within it. We treat the normalized saliency map as a distribution and use it to sample a large number of random points. These points are clustered using mean-shift [7]. The centers of the clusters are the locations of our candidates. Finally, we estimate the covariance matrix of each candidate by fitting a Gaussian to the saliency map in the neighborhood of the candidate location. The neighborhood size is set to $1/5$ of the frame height, to avoid interference with other candidates. We intentionally do not use Gaussian mixture model, since we prefer capturing the peaks over the broader contours of the distribution.

An example of our static candidates is provided in Figure 2,(a). Candidates were created around the most salient regions of the image, such as the face and the label on the back. Their size reflects the size of the region. Furthermore, some candidates capture less salient regions, such as the two bars.

4.2. Motion candidates

Modeling the saliency in independent frames is insufficient for videos since it ignores the dynamics. It is well known that motion attracts human attention [17]. Thus, next we incorporate motion cues into our salient candidate set.

To produce motion candidates we first calculate the optical flow between consecutive frames [22]. We keep only the optical flow magnitude and filter out pixels with weak flow as unreliable. Since we are interested in local motion contrast we apply Difference-of-Gaussians (DoG) filtering to the optical flow magnitude. The motion candidates are created from the DoG map in the same way as the static candidates are created from the image saliency map (i.e., mean-shift clustering and Gaussian fitting).

An example of the resulting motion candidates is illustrated in Figure 2,(b). In this frame of the video the man bends his arm and moves a brush to paint the wall. Accordingly, motion candidates were detected at the brush and elbow.

4.3. Semantic candidates

Finally we wish to add semantic candidates to our set. These candidates represent regions that attract human attention due to higher level visual processing or other priors. We consider three types of semantic candidates.

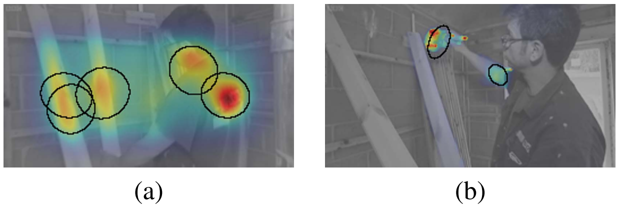


Figure 2. **Static (a) and motion (b) candidates.** The original frame is shown in gray (for visualization) It is overlaid with: (a) the GBVS saliency map and (b) optical flow magnitude. The candidates are indicated by black ellipses, that mark a radius σ of the corresponding Gaussian.

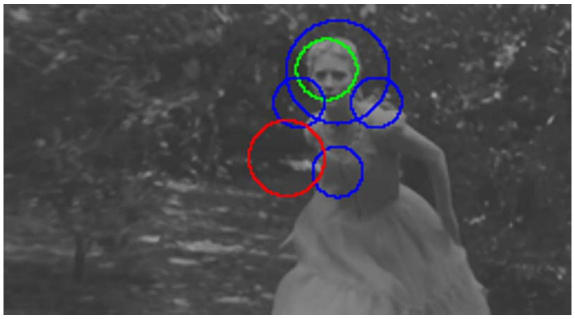


Figure 3. **Semantic candidates.** The center candidate is red, green ellipses are faces, and blue are human body candidates. Since the body is large it is represented by four candidates. These candidates cover most of the semantically salient regions in the frame.

First, it has been shown that humans watching a video are biased towards the center of the screen [31]. Therefore, we create a constant size *center candidate* at the center of the frame.

People are also known to fixate on faces (when they are large) and on the torso (in more distant shots) [18]. To detect these we run a face detector [4] and a poselet detector [5] on the frame of interest. These provide the location and the size of the faces and the bodies. Additionally, we run a mean shift non-maximal suppression of the detected bounding boxes to prevent overlapping duplicate detections.

Since the detectors find faces and bodies at different scales we treat large and small detections differently. First, detections with very small bounding boxes (less than 15% of the frame height) are rejected as noisy. For the remaining small detections we create a single candidate at their center. For large detections we create several candidates: four for body detections (head, shoulders and torso) and three for faces (eyes and nose with mouth). The placement of the candidates is fixed inside the detected bounding box and the covariance is proportional to the size of the bounding box.

All three types of candidates – center, face, and body – are illustrated in Figure 3.

5. Modeling gaze dynamics

Having extracted a set of candidates we next wish to select the most salient one. We accomplish this by learning *transition probability* – the probability to shift from one gaze location in a source frame to a new one in a destination frame. This transition is different from a saccade – we are dealing with a shift of the entire distribution, while a saccade is a rapid movement of a gaze point. Note that the source frame is not necessarily the immediately preceding frame, but can be several frames earlier in time. This allows us to model the gaze dynamics in the video and predict the saliency more accurately.

5.1. Features

To model changes in focus of attention we associate a feature vector with pairs of source and destination candidates in a given pair of frames. In the following we describe the creation of a feature vector for every ordered pair of (source, destination) candidate locations.

The features can be categorized into two sets: destination frame features and inter-frame features. We experimented with the use of source frame features as well, but found these features led to overfitting in the learning process, as they are only slightly different from the destination frame features. We use static, motion and semantic features, as described next.

As a low level spatial cue we use the local contrast of the neighborhood around the candidate location. The local contrast is computed as:

$$C_l = \frac{I_n^{max} - I_n^{min}}{(I_n^{max} + I_n^{min}) \cdot C_g}, \tag{1}$$

where I_n^{min}, I_n^{max} are the minimum and the maximum intensity in the local neighborhood. C_g is a global contrast scale calculated as:

$$C_g = \frac{I^{max} - I^{min}}{I^{max} + I^{min}}, \tag{2}$$

where I^{min}, I^{max} are the minimum and the maximum intensity of the frame. Additionally, we compute the mean GBVS of the candidate neighborhood and add it to the set of features.

To represent local motion we first compute the Difference-of-Gaussians (DoG) of the vertical and horizontal components of the optical flow as well as for its magnitude. We then add to the feature vector the mean value of every DoG map in the local neighborhood of the destination candidate.

Finally, we add a set of semantic features. First, we add face and person detection scores (as described in Appendix A). We further add discrete candidate labels: motion, saliency, face, body, center, and the size of the corresponding region. To account for the center bias we use the Euclidean distance from the candidate location to the center of the frame. It is important to note that all types of features are computed for all the destination candidates regardless of the type of the candidate.

5.2. Gaze transitions for training

We pose the learning problem as classification: whether a gaze transition occurs from a given source candidate to a given target candidate. To train such a classifier based on the features described in the previous section we need (i) to choose relevant pairs of frames, and (ii) to label positive and negative gaze transitions between these frames.

To choose a set of relevant frames we use the most obvious places for attention shifts – scene cuts. We find all the cuts in the training set using a scene cut detector [32] and set the source frame to be the last frame immediately preceding the cut. Since it takes 5 to 10 frames for humans to fixate on a new object of interest we set the destination frame 15 frames after the cut [13]. This ensures that we will not learn from incomplete or partial gaze transitions. For negative samples we choose pairs of frames from the middle of every scene. For consistency we set the gap between the source and destination to be 15 frames.

Next, we need to obtain examples of positive and negative gaze transitions. We start by aggregating the ground truth human fixations into clusters (for both source and destinations frames). This is done by smoothing the fixation maps and thresholding them to keep only the top 3%. This provides a set of distinct regions of attention. We consider the centers of these regions as foci of attention. The foci of the source frame are taken as source locations. We take all pairs of source locations and destination candidates for our training set. Pairs with a destination candidate near a focus of the destination frame are labeled as positive. All other pairs are labeled negative. We illustrate the labeling in Figure 4.

5.3. Learning transition probability

The last stage of our system learns the classifier for whether a transition occurs or not, i.e., the probability of each pair of source-destination transition. We first calculate the mean of each feature and its standard deviation across the training set. Each feature is normalized to have zero mean and unit standard deviation. The normalization parameters are stored together with the trained classifier.

We train a standard random forest classifier [21] using the normalized feature vectors and their labeling. At the inference stage the trained model classifies every transition between source and destination candidates and provides a confidence value. We use the normalized confidence as the transition probability $P(d|s_i)$ – the transition probability from the source s_i to the current destination candidate d. By aggregating all the transitions together we get the final

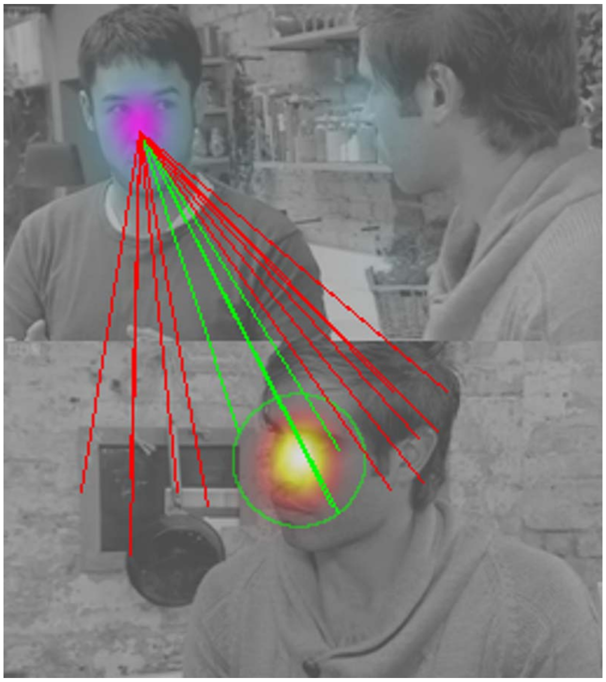


Figure 4. **Positive and negative examples of gaze shifts.** The green (positive) and red (negative) lines mark pairs of possible source-destination transitions. The transition pairs are overlayed on the source (top) and destination (bottom) frames, together with source (magenta) and destination (yellow) gaze maps.

probability of the candidate as:

$$P(d) = \sum_{i \in S} P(d|s_i) \cdot P(s_i) \tag{3}$$

where

$$P(s_i) = \frac{Sal(s_i)}{\sum_{i \in S} Sal(s_i)} \tag{4}$$

and $Sal(s_i)$ is the source candidate saliency and S is the set of all the sources. Finally, we produce the saliency map in a similar fashion to how Gaussian mixture models are used to create a continuous distribution: we replace each candidate with a Gaussian of corresponding covariance and sum them up using the candidate saliency as weight.

6. Experimental validation

In this section we experimentally validate the proposed video saliency detection method. For our experiments we use the DIEM (Dynamic Images and Eye Movements) dataset [24], which includes 84 high-definition videos from different styles, such as movie trailers, ads, sport events, etc. Most of the videos are professionally produced and the image quality is excellent. The dataset is provided together with gaze tracks of about 50 participants per video.

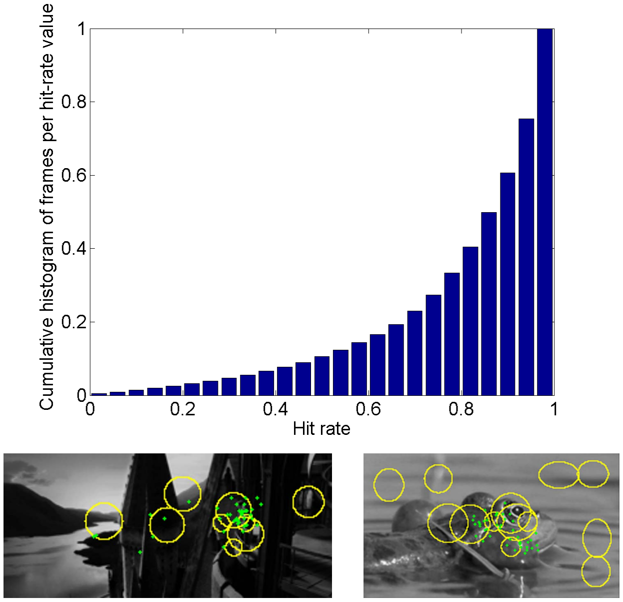


Figure 5. **Our candidates cover most human fixations.** (a) Cumulative histogram of per-frame hit-rate of fixation points inside the candidates. It can be seen that most of the fixations are captured well by the candidates. (b), (c) Example frames, together with human fixation points (green) and our extracted candidates (yellow). Our candidates cover most of the fixation points.

6.1. Verification of the candidates

First, we wish to demonstrate that human fixations can be modeled well by our limited candidate set. To do so we count the number of candidate locations that are "close enough" to a fixation point. If a fixation point falls inside the ellipse defined by the candidate's covariance matrix, we call it a hit. To define the ellipse we threshold the corresponding Gaussian at a radius of σ. Otherwise, it is a miss. The histogram of the hit rate over all the frames in DIEM is shown in Figure 5 (a). The average hit rate over all the frames is 81%, and the median is 88%. This means that on most of the frames most of the fixations can be modeled well by our candidate set. Additionally, Figures 5 (b),(c) show a visual comparison between the human fixations and our candidates.

6.2. Performance evaluation

To evaluate the accuracy of the proposed method we follow the train / test scheme proposed by Borji et al. [3]. The test set includes all frames of 20 representative videos. The model is trained on the remaining 64 videos. Since our method computes the probability to shift from a location in a source frame to a location in a destination frame, we calculate the video saliency in a sequential order. For the first frame of the video we use as source a single location at the center. For every following frame we compute transition probability to its candidate set using the predicted saliency map from the previous frame as the source. This method does not drift over time, since the transitions are largely independent of the source frame properties (recall that features of the source frame were excluded and the destination candidates are computed independently for each frame).

6.2.1 Evaluation procedure

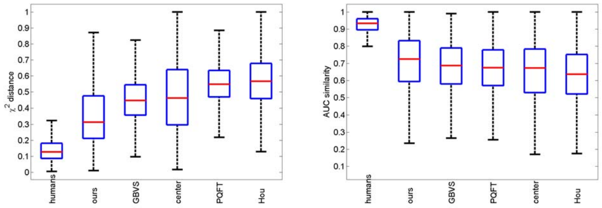
We use two different metrics to quantitatively evaluate performance. The first metric is the area-under-curve (AUC), which utilizes the receiver-operator curve to compute the similarity between human fixations and the predicted saliency map. We use its shuffled variation which accounts for central bias [25]. The higher the AUC, the better the result is.

Since the AUC considers the saliency results only at the locations of the ground truth fixation points, it cannot distinguish well between a peaky saliency map and a smooth one. In other words, the AUC considers each fixation separately rather than viewing the fixations as samples of a distribution. Thus we propose an additional metric for performance evaluation, the well-known χ^2 distance between two distributions. The χ^2 distance will prefer a peaky saliency map over a broad one, when comparing them to the tight distribution of the ground truth. For χ^2 a lower value implies a better result.

We convert the sparse ground truth fixation map, recorded by the gaze tracker, to a dense probability map by convolving it with a constant size Gaussian kernel. The size of the kernel is set experimentally to maximize the self-explanation of the points over all video frames. This means that we set the kernel size (11 pixels) so that the "humans" measure described below is minimized under the χ^2 metric.

We compare the proposed saliency prediction approach with five different methods. The first, referred to as *humans*, serves as an upper bound for the saliency prediction and measures how much the fixation map explains itself. To calculate it we randomly split the ground truth fixations of a frame into two halves and compare the distributions created from each half using χ^2. The distributions are created by convolving with the mentioned Gaussian kernel. We repeat this 10 times and average the result.

The second method we compare to is a Gaussian placed in the center of the frame. This method is used by Judd et al [18] and we follow the parameters from there. We further compare our results to the image saliency approach of GBVS [12], and two video saliency methods PQFT [11] and the method of Hou and Zhang [14] (annotated in figures and tables as Hou for brevity). Both methods are among the highest rated video saliency algorithms according to the recent benchmark of Borji et al.[3].



(a) (b)

Figure 6. **Our algorithm outperforms all others on DIEM.** The plots compare overall performance on DIEM dataset. We show both χ^2 distance (a), in which the lower the result the better, and the AUC (b), for which the higher the better. The red lines show the median and the blue boxes represent the 90-th percentile.

6.2.2 Results

First, we evaluate the performance of our approach without the proposed candidate selection. That is, we applied the entire procedure while considering all the pixels as candidates, rather than using our selected set of candidates. Using dense computation is orders of magnitude slower, rendering it impractical. Furthermore, dense estimation resulted in lower accuracy than the candidate based approach. Using χ^2 measure we get 0.347 median distance over our testing set, compared to 0.313 when using candidate selection (recall that for χ^2 lower scores mean better results). Since our candidate-based approach is both more accurate and more efficient, we use it for all our following experiments.

A comparison of our candidate-based approach with the aforementioned other algorithms is presented in Figure 6. In this plot the median result over all the frames is marked by a red line, and the blue boxes represent the 90-th percentile of the similarity. Our method outperforms all other algorithms in both AUC and χ^2 measures. Using χ^2 further emphasizes the benefits of our approach: we produce a tight distribution that is more similar to the original gaze map.

To evaluate the contribution of each type of cue we perform leave-one-cue-out experiments. We train and test the same system while excluding one of the four cue types: static, motion, semantic and inter-frame. We use the same DIEM dataset as in the previous experiment and the same splitting. Table 1 presented the median results of the χ^2 measure relative to the proposed complete measure that include all cues (the lower the better). As can be seen, dropping any cue type decreases the performance. The most significant cues for the system are static and semantic cues. Furthermore, if all semantic cues are removed from the learning procedure our proposed approach still outperforms the nearest competitor (that does not use semantic information).

We further visually compare our saliency maps to those of other methods. Figure 7 demonstrates several examples (the complete video results can be found in the supplementary). As can be seen, the saliency maps produced

Table 1. **Per-cue analysis of our algorithm on DIEM.** As one can see dropping static or semantic cues considerably decreases the performance.

	All cues	No motion cues	No inter-frame cues	No semantic cues	No static cues
χ^2	0.313	0.322	0.326	0.347	0.385

Table 2. **Our algorithm outperforms all others on CRCNS.** As can be seen, our method provides lowest χ^2 distance to the ground truth, rendering it the best performing approach.

	Humans	Ours	Center	GBVS	PQFT	Hou
χ^2	0.42	0.43	0.53	0.51	0.61	0.63

by the proposed method are more visually consistent with the shape, size, and location of the ground truth gaze map than the maps of the other methods.

In addition we also experimented on the CRCNS dataset [15]. It includes 50 low quality videos of VGA resolution. The ground truth fixation data for the set is collected from 8 participants. As for DIEM, we choose a testing set of 12 videos and put the others in the training. The testing videos are chosen to represent different video categories. As shown in Table 2, our method outperforms all other algorithms on this dataset as well.

7. Conclusions

In this paper we proposed a novel method for video saliency prediction. The method is substantially different from existing methods and uses a sparse candidate set to model the saliency map. It is shown experimentally that using candidates boosts the accuracy of the saliency prediction and speeds up the algorithm. Furthermore, the proposed method accounts for the temporal dimension of the video by learning the probability to shift between saliency locations.

A. Implementation details

Here we present some implementation details of our approach. We use the same settings in all our experiments.

We downsample all videos to 144 rows using bilinear interpolation while preserving the aspect ratio. When determining the motion candidates we filter out all regions with optical flow magnitude lower that 2 pixels. For the Difference-of-Gaussians filtering we use $\sigma = 10$ and $\sigma = 20$ pixels. For the center candidate we set $\sigma = 1/8 \cdot FrameHeight = 18$ pixels. In the face and body candidates the size depends on the height of the detection. If the detection is smaller than $0.4 \cdot FrameHeight$ it is considered a small target and modeled using a single candidate with $\sigma = 1/3 \cdot DetectionHeight$. Larger detections of body create three candidates (torso and shoulders) with $\sigma = 1/12 \cdot DetectionHeight$ and one candidate for the head with $\sigma = 1/6 \cdot DetectionHeight$ (their layout is depicted in Fig. 3). Larger detections of faces are modeled by three candidates with $\sigma = 1/4 \cdot DetectionHeight$ (eyes and mouth).

When calculating static and motion features in the neighborhood of a candidate we use three different neighborhoods, sized 5×5, 9×9 and 17×17 pixels. As we calculate the semantic features we aggregate the face and human detections together. This is accomplished by replacing each detection with a Gaussian with σ corresponding to the detection size. After summing all the Gaussians together we sample the maps at candidate locations to create this feature.

Acknowledgments

We are grateful to Adobe for a wide support of this research. We thank the DIEM database for making the gaze tracking results publicly available. This research was funded (in part) by the Intel Collaborative Research Institute for Computational Intelligence (ICRI-CI), the Ollendorf foundation and by the Israel Ministry of Science.

References

[1] B. Block. *The visual story: seeing the structure of film, TV, and new media*. Focal Press, 2001.

[2] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *PAMI*, 2012.

[3] A. Borji, D. Sihite, and L. Itti. Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study. *IEEE Transactions on Image Processing*, 2012.

[4] L. Bourdev and J. Brandt. Robust object detection via soft cascade. In *CVPR*, pages 236–243, 2005.

[5] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, pages 1365–1372, 2009.

[6] G. Buswell. How people look at pictures: a study of the psychology and perception in art. 1935.

[7] Y. Cheng. Mean shift, mode seeking, and clustering. *PAMI*, 17(8):790–799, 1995.

[8] X. Cui, Q. Liu, and D. Metaxas. Temporal spectral residual: fast motion saliency detection. In *Proceedings of the ACM international Conference on Multimedia*, 2009.

[9] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. *PAMI*, 34(10):1915–1926, 2012.

[10] R. Goldstein, R. Woods, and E. Peli. Where people look when watching movies: Do all viewers look at the same place? *Computers in biology and medicine*, 37(7):957–964, 2007.

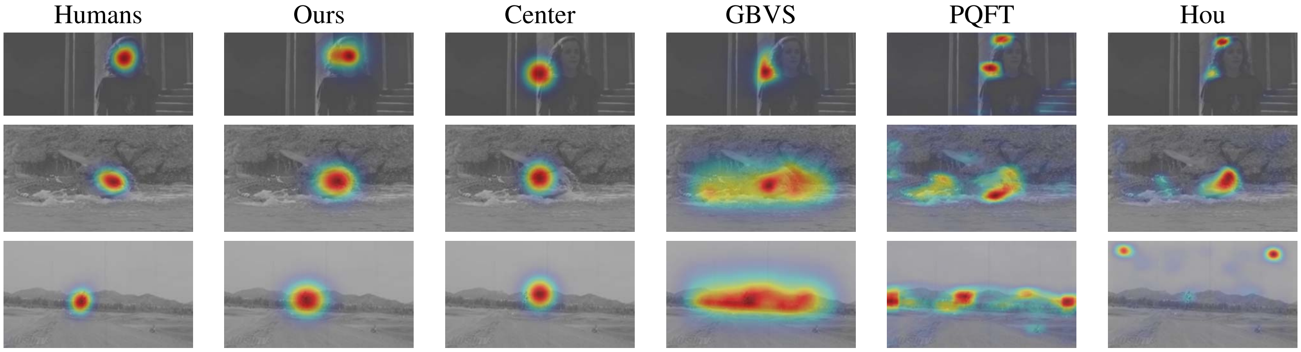


Figure 7. **Our saliency maps resemble the ground truth.** Examples of saliency detection results using different methods show that the saliency predicted by the proposed method better approximates the human gaze map.

[11] C. Guo, Q. Ma, and L. Zhang. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In *CVPR*, pages 1–8, 2008.

[12] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. *NIPS*, 19:545, 2007.

[13] J. Henderson. Human gaze control during real-world scene perception. *Trends in cognitive sciences*, 7(11):498–504, 2003.

[14] X. Hou and L. Zhang. Dynamic visual attention: Searching for coding length increments. *NIPS*, 21:681–688, 2008.

[15] L. Itti. Automatic foveation for video compression using a neurobiological model of visual attention. *IEEE Transactions on Image Processing*, 13(10):1304–1318, 2004.

[16] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *PAMI*, 20(11):1254–1259, 1998.

[17] G. Johansson. Visual perception of biological motion and a model for its analysis. *Perceiving events and objects*, 1973.

[18] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *ICCV*, pages 2106–2113, 2009.

[19] W. Kim, C. Jung, and C. Kim. Spatiotemporal saliency detection and its applications in static and dynamic scenes. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(4):446–456, 2011.

[20] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4(4):219–27, 1985.

[21] A. Liaw and M. Wiener. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[22] C. Liu. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Massachusetts Institute of Technology, 2009.

[23] V. Mahadevan and N. Vasconcelos. Spatiotemporal saliency in dynamic scenes. *PAMI*, 32(1):171–177, 2010.

[24] P. Mital, T. Smith, R. Hill, and J. Henderson. Clustering of gaze during dynamic scene viewing is predicted by motion. *Cognitive Computation*, 3(1):5–24, 2011.

[25] B. Schauerte and R. Stiefelhagen. Predicting human gaze using quaternion dct image signature saliency and face detection. In *IEEE Workshop on Applications of Computer Vision (WACV)*, pages 137–144. IEEE, 2012.

[26] H. Seo and P. Milanfar. Static and space-time visual saliency detection by self-resemblance. *Journal of Vision*, 9(7), 2009.

[27] T. Smith. Attentional theory of cinematic continuity. *Projections: The Journal for Movies and Mind*, 6(1):1–27, 2012.

[28] T. Smith and J. Henderson. Edit blindness: The relationship between attention and global change blindness in dynamic scenes. *Journal of Eye Movement Research*, 2(2):6, 2008.

[29] Tobii. Advertising research and eye tracking. `http://www.tobii.com/eye-tracking-research/global/research/advertising-research/`.

[30] A. Treisman and G. Gelade. A feature-integration theory of attention. *Cognitive psychology*, 12(1):97–136, 1980.

[31] P. Tseng, R. Carmi, I. Cameron, D. Munoz, and L. Itti. Quantifying center bias of observers in free viewing of dynamic natural scenes. *Journal of Vision*, 9(7), 2009.

[32] S. Zanetti, L. Zelnik-Manor, and P. Perona. A walk through the webs video clips. In *CVPRW*, pages 1–8. IEEE, 2008.