Probabilistic Learning of Task-Specific Visual Attention

Ali Borji Dicky N. Sihite Laurent Itti
Department of Computer Science, University of Southern California, Los Angeles
http://ilab.usc.edu

Abstract

Despite a considerable amount of previous work on bottom-up saliency modeling for predicting human fixations over static and dynamic stimuli, few studies have thus far attempted to model top-down and task-driven influences of visual attention. Here, taking advantage of the sequential nature of real-world tasks, we propose a unified Bayesian approach for modeling task-driven visual attention. Several sources of information, including global context of a scene, previous attended locations, and previous motor actions, are integrated over time to predict the next attended location. Recording eye movements while subjects engage in 5 contemporary 2D and 3D video games, as modest counterparts of everyday tasks, we show that our approach is able to predict human attention and gaze better than the state-of-the-art, with a large margin (about 15% increase in prediction accuracy). The advantage of our approach is that it is automatic and applicable to arbitrary visual tasks.

1. Introduction

Visual attention is an important facet of our vision in everyday life. It makes processing complex visual scenes tractable through sequential selection of localized image regions. It is commonly believed that visual attention is guided by two components: 1) a bottom-up (BU), task-independent, and image-based component that instinctively draws the eyes to places in the scene that contain discontinuities in image features, such as motion, color, and texture, and 2) a top-down (TD) component that guides attention and gaze in a task-dependent and goal-directed manner, orchestrating the sequential acquisition of information from the visual environment. In everyday life, these two components are combined in the control of gaze.

In computer vision, research on visual attention has been primarily focused on the BU component. Early studies were directly influenced by cognitive studies of visual search and Feature Integration Theory (FIT) [12]. This led Koch and Ullman [13] to define the saliency map: A topographic map with retinotopic organization where locations that stand out in an image (e.g., because of distinctive features such as color, texture, and motion) are highlighted. The first complete implementation and verification of this architecture was done by Itti *et al.* [7]. Several other approaches for detecting image-based outliers have also been proposed, based on information theory [14], discriminant hypothesis [15], spectral models [16], sparse and efficient coding [17], and Bayesian and graphical models [18][19][21].

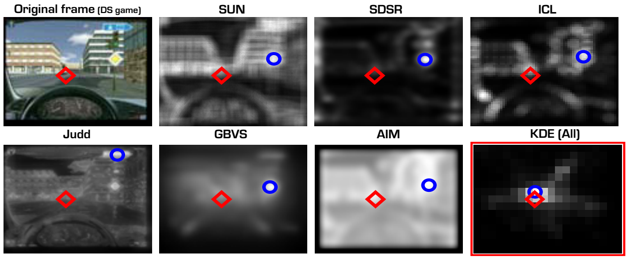

Figure 1. Bottom-up saliency does not account for task-driven eye movements. Predictions of 6 state-of-the-art BU saliency models in a driving scene and our model (red box). Red diamond and blue circle show human fixation and maximum of a model, respectively. See Table 2 for results.

Today, saliency detection and eye movement prediction over static images and videos is a reasonably well-researched area and there are many models with good accuracy, although of course improving tolerance to noise and invariance of algorithms is always possible. However, success of BU models is limited to a small range of everyday tasks, such as free-viewing [7][14][18] and their adaptation to visual search [22][11]. BU models usually can not predict exact fixation points and leave more than half of them unaccounted [25] (Fig. 1). One problem with models based on saliency maps is that they are correlated with fixation behaviors but don't tell much about the cause for such behaviors [26][27]. Diverging from the current trend, we focus on modeling top-down attention which can boost performance of several approaches in computer vision. For example in areas such as object detection and recognition [15][35][36], especially in spatio-temporal domain for video understanding and action/event recognition (e.g., [37][38]).

We aim to build an attentive vision system that can tell where it should look as it moves through the world and interacts with the environment. This problem is very important but very difficult and largely unsolved. Our approach is to utilize global visual context [28][18], a low-dimensional representation of the whole image (the "gist" of the scene).

978-1-4673-1228-8/12/$31.00 ©2012 IEEE

Such a representation can be easily computed, and it relaxes the need to identify specific regions or segment and recognize all objects in a scene. We focus on interactive environments (contemporary 2D and 3D video games), where visual stimuli are dynamically generated and affected by deliberative motor actions. We develop top-down models trained over the same or similar games from data of subjects during game playing, and we use those models to predict saccades of a new test subject. Several sources of multi-modal information, including global context, previous saccades, and previous motor actions, are combined in a unified Bayesian framework over time. Compared with brute-force algorithms such as the average of all saccade positions, a central Gaussian blob, and 6 popular BU models, we show that our models significantly outperform the state-of-the-art in terms of accuracy at predicting where a new subject looks during active gameplay. This indicates the effectiveness of our approach for modeling complex task-driven attention.

Previous Work. The majority of studies on TD attention are at the analysis/descriptive level and there are few computational models available, we believe due to conceptual complexity. Yarbus [1] discovered a compelling finding that 'seeing' is inextricably linked to the observer's cognitive goals. Task dependency of gaze has been extensively studied for several real-world tasks, such as "sandwich making" [4], "tea making" [2], and "driving" [5]. These studies have revealed that most fixations are directed to task-relevant locations, and there is a tight temporal relationship between fixations and task-related behaviors, such that it is sometimes possible to infer the algorithm of a task from the pattern of a subject's eye movements (e.g., in "block copying" [4]). In [3], Hayhoe and Ballard elaborate on the role of internal reward in guiding eye and body movements, supported by neurophysiological studies. Inspired by the idea of visual routines [29] and using reinforcement learning (RL) approaches, Sprague and Ballard [9] proposed an RL-based top-down attention model for explaining eye movements of an agent operating in virtual environments. This approach is interesting but suffers from three limitations that make it hard to apply directly for computer vision purposes. First, it is limited to laboratory-scale tasks such as side-walk navigation [9], second, visual processing is very simple, and, third, it needs explicit definitions of reward functions, subtasks, and arbitration mechanisms.

Our approach has in part similarities with the contextual model of Torralba *et al.* [18] as we also use the concept of gist. We start with a basic Bayesian formulation and add new features to account for task-driven attention in spatio-temporal domain while former has been thus far utilized for bottom-up saliency and visual search over static stimuli. This model and its decedents (e.g., [19, 11]) originally formulate object search as estimating the probability $P(O = 1, X|L, G)$ where $X = (x, y)$ defines the location of the target in the image, O is a binary variable ($O = 1$ denotes target presence and $O = 0$ denotes target absence in the image), and L and G denote local and global features, respectively. According to Bayes' theorem, they expand the above probability as:

$$
\begin{aligned}
&P(O = 1, X|L, G) = \\
&\frac{1}{P(L|G)} P(L|O = 1, X, G) P(X|O = 1, G) P(O = 1|G)
\end{aligned}
\quad (1)
$$

The first factor on the right, $\frac{1}{P(L|G)}$, is independent of the target, measures BU saliency, and is solely dependent of local image features. The second factor represents top-down knowledge of target appearance. Image regions with features likely to belong to the target object are enhanced. The third factor provides context-based priors on the location of the target, and the fourth factor provides the prior probability of presence of the target in the scene. If this probability is very small, then object search need not be initiated. Zhang *et al.* [19] used Independent Component Analysis (ICA) and Difference of Gaussians (DOG) features learned from a large repository of natural scenes to estimate the first factor. From another perspective these models unify the information theoretic models (e.g., [14]), in the sense both are based on self-similarity of scene regions. These models assign higher saliency values to regions with rare features. Information of visual feature F is $I(F) = -log\ P(F)$ which is inversely proportional to the likelihood of observing F. By fitting a distribution $P(F)$ to features, rare features can be immediately found by computing $P(F)^{-1}$ in an image. The idea of global context has also been extensively employed in several areas of computer vision (e.g., [32][10]).

Several other approaches have been proposed to model top-down attention, specifically for visual search. Navalpakkam and Itti [22] proposed a Bayesian approach to derive the optimal gains that should be applied to low-level visual features contributing to a saliency model [7], to make an object of interest more salient. The objective was to maximize the signal to noise ratio of the expected target object versus background clutter, and training was performed over a set of natural scenes containing ground-truthed objects. An intuitive solution for the same problem (optimal gains of feature channels) was suggested earlier by Frintrop [23] which is the end result of the SNR maximization process in [22]. Navalpakkam and Itti [8] proposed conceptual guidelines for modeling the role of task on visual attention, but their method requires the algorithm of the task to be known, and is not fully implemented.

Perhaps the most similar work to ours (i.e., real-world and unconstrained tasks) is the work by Peters and Itti [6], where they used gist as a predictor of fixation, learning from examples where people looked in scenes of different gists and while engaged in a particular task. The same scene gist, however, might not always warrant the same eye movement, based on the history and sequence of previous fixations and actions to date. For example, in one of the games studied here, even when looking at the exact same scene, eye movements are often guided by past events, such as different customers placing different orders for items which the player is asked to provide. To tackle the problem that gist of the scene is not enough, we follow a sequential processing framework where several factors predictive of eye movements are integrated over time and can resolve the confusion (aliasing) at one snapshot of time.

2. Proposed Model

Our goal is to predict where a human subject attends under the task influence T. This is similar to explaining saccades (jumps in eye movements) in free-viewing, addressed by bottom-up models, with the difference that here a policy governs saccades. Since it is difficult to learn general strategies for performing every task, here we focus on learning models for each task separately. Following a leave-n-out approach over subjects, first, in the training phase, we compile a training set of feature vectors and eye positions corresponding to individual frames from several video game clips which were recorded while observers were playing video games. Then, training data is used to learn probability distributions over image locations for given feature vectors, and pdfs are later leveraged in the test phase for inferring the next attended location of a new test subject.

We need a number of variables that cause or correlate with saccade positions and hence can provide information regarding the next saccade location. These variables tell us indirectly about the state of the agent at each time point of the task. In addition to scene gist, here, we introduce two new features explained below: motor actions and previous saccade position and then combine them in a probabilistic manner over time to infer a probability distribution over scene locations that may attract next saccade:

Global context (Gist, G). Following a brief presentation of a photograph, humans are able to summarize the quintessential characteristics of an image, a process previously expected to require much analysis. A number of models exist for calculating Gist (e.g., [18][33]). We adopt the gist model of [10][1] as it is based on the bottom-up saliency model [7] that we use here as a baseline approach. We consider 4 scales for each orientation pyramid, 6 scales for each color pyramid, and 6 scales for intensity. For each of the maps, average in each of the patches of grid sizes $n \times n$ (here $n \in \{1, 2, 4\}$) are calculated (thus 21 values). Overall the final gist vector will be augmentation of $(4 \times 4 + 6 \times 2 + 6 \times 1) \times 21 = 714$ values. We then employ PCA to reduce the dimensionality. We also, investigate the ability of histogram of oriented gradient (HOG) [30] features to represent the global context of a scene[2].

Previous saccade location (X). A lot of everyday tasks need a number of perceptions and actions to be performed in a sequence (e.g., sandwich making [4]). Therefore, knowing what object has been attended previously gives an evidence for the next attended object. We implement this idea over spatial locations. For instance, $P(X^{t+1} = b|X^t = a)$ indicates the probability of looking at location b in the next time step given that location a is currently fixated (e.g., looking at left first and then right, when turning right).

Motor actions (A). Actions and fixations are tightly linked thus, by knowing a performed action, one can tell where to look next. We recorded motor actions while humans were involved in game playing. We assumed that these actions correspond to some high-level events in the game (e.g., mouse click for shooting). We logged actions for driving games (e.g., wheel position, pedals (brake and gas), left and right signals, mirrors, left and right side views, and gear change), from which we only generated a 2D feature vector from wheel and pedal positions between 0 and 255 (Fig. 2). For other games, 2D mouse position and joystick buttons were used (further explained in Sec. 3.1).

2.1. Problem Formulation

In this section, we describe details of our Bayesian approach of information integration over time to predict saccade in the next time step. Our method is based on Hidden Markov Models (HMM), which are successful probabilistic tools for sequence processing. We are particularly interested in the probability of attending to spatial location X given all available information I, or $P(X|I)$. One way to estimate this probability is to follow a discriminative approach by augmenting all information into a large vector I, and using a classifier to map it to X from a set of labeled training data. An alternative is to follow a Bayesian formulation: $P(X|I) = P(I|X)P(X)/P(I) = \mu P(I|X)P(X)$. Parameter μ is selected in a way that resultant probabilities sum to 1 (i.e., $\sum_j P(X_j|I) = 1$). $P(X)$ is simply the prior distribution of all saccade locations in the training data (sum of all saccades or average fixation map). A benefit of the generative approach over the discriminant classifier-based approach is that, it provides a unified method for information integration of sequential data, and makes it suitable for our purpose. which enhances results.

Formally, the goal of the saccade prediction is to compute a probability distribution over the possible locations given all features up to time t. Let $X_t \in \{1 \ldots n\}$ denote the saccade location with n as the number of locations in the image at time t. To generate sufficient data, we resize the original eye fixation map with one at the attended location and zeros elsewhere into a smaller scale map (a $w \times h$ grid). Therefore, X_t is the location of 1 in such map. In the following we start with the simplest case of $P(X|I)$ when only global context information is available (i.e., I is equal to Gist) and add more information in subsequent steps.

[1] http://ilab.usc.edu/siagian/Research/Gist/Gist.html

[2] http://pascal.inrialpes.fr/soft/olt/

Case 1: Gist only. In this case, only global context information from all past and the current time is used. According to the Bayes theorem we have:

$$\begin{aligned} P(X_t|G_{1:t}) &= P(X_t|G_t, G_{1:t-1}) \\ &= \frac{P(G_t|X_t)P(X_t|G_{1:t-1})}{P(G_t|G_{1:t-1})} \\ &= \mu P(G_t|X_t)P(X_t|G_{1:t-1}) \end{aligned} \tag{2}$$

Following Markov assumption, the current scene Gist (G_t) has all the necessary information for determining state and knowing the attended location. Thus X_t is independent of all previous gists: $P(X_t|G_{1:t-1}) = P(X_t)$. Therefore, we can write: $P(X_t|G_{1:t}) = \mu P(G_t|X_t)P(X_t)$ with $P(X_t)$ as the prior distribution over eye positions.

Case 2: Gist and previous saccade. In the second step, we add the previous saccade locations to the formulation:

$$\begin{aligned} & P(X_t|G_{1:t}, X_{1:t-1}) \\ &= P(X_t|G_t, G_{1:t-1}, X_{1:t-1}) \\ &= \frac{P(G_t|X_t)P(X_t|G_{1:t-1}, X_{1:t-1})}{P(G_t|G_{1:t-1}, X_{1:t-1})} \\ &= \mu_1 P(G_t|X_t)P(X_t|G_{1:t-1}, X_{1:t-1}) \\ &= \mu_1\mu_2 P(G_t|X_t)P(X_{t-1}|X_t)P(X_t|G_{1:t-1}, X_{1:t-2}) \end{aligned} \tag{3}$$

where μ_1 is equal to $P(G_t|G_{1:t-1}, X_{1:t-1})^{-1}$ and μ_2 is $P(X_{t-1}|G_{1:t-1}, X_{1:t-2})^{-1}$. Again, considering Markov assumption and defining $\mu = \mu_1\mu_2$, we have: $P(X_t|G_{1:t}, X_{1:t-1}) = \mu P(G_t|X_t)P(X_{t-1}|X_t)P(X_t)$.

Case 3: Gist, previous saccade, and motor actions. Finally, we combine all evidences in our Bayesian model. Following the steps in case 2 and simplifying we reach to:

$$\begin{aligned} & P(X_t|G_{1:t}, X_{1:t-1}, A_{1:t-1}^{j=1:n}) \\ &= \mu P(G_t|X_t)P(X_{t-1}|X_t)P(X_t) \times \Pi_{j=1}^{n} P(A_{t-1}^{j}|X_t) \end{aligned} \tag{4}$$

The above formula assumes that actions are independent of each other given attended location (i.e., $A^k \perp A^l | X$). An important point here is whether actions influence saccades or vice-versa. In the real world the interaction works both ways: for some situations/tasks, saccades lead actions, however, sometimes actions can also lead eye movements. Here to be on the safe side, we did not use the current action.

Computing (4) requires estimation of $P(G_t|X_t)$ and similarly others. This can be done in several ways using non-parametric probability density estimation techniques such as generalized Gaussian model, histogram estimation or kNNs. We adapted the Kernel Density Estimation (KDE) approach. One pdf is calculated for each spatial location:

$$P(G|x_i) = \frac{1}{m}\sum_{i=1}^{m} \mathcal{G}_h(x - x_i) = \frac{1}{mh}\sum_{i=1}^{m} \mathcal{G}(\frac{x - x_i}{h}) \tag{5}$$

where $\mathcal{G}_h$ is a Gaussian kernel with smoothing parameter (sliding window or bandwidth) h and m is number of data points. We used a Matlab toolbox[3] for implementing KDE.

2.2. Baseline Benchmark Models

To fully evaluate effectiveness of our model, we implemented the regression model put forward by Peters and Itti [6] as well as a nearest-neighbor classifier and two other brute-force yet powerful models.

Linear Regression (REG). This model does not take into account the temporal progress of a task and simply maps Gist of the scene to the eye position. Mathematically, the goal is to optimize the following objective function:

$$\begin{aligned} & \arg\min_{W} ||M \times W - X_{sacc}||^2 \\ & Subject\ to: W \geq 0. \end{aligned} \tag{6}$$

where M indicates the matrix of feature vectors (only Gist feature is used in [6]) and X is the matrix of eye positions (one fixation per frame). The least-squares solution of the above objective function is: $W = M^+ \times X$, where M^+ is the pseudo-inverse of matrix M through SVD decomposition. In our experiments, we only take the largest eigenvalue of the SVD since this avoids numerical instability and results in higher accuracy. Given vector $E = (u, v)$ as the eye position over a 20×15 map (i.e., $w = 20, h = 15$) with $u \in [1, 20]$ and $v \in [1, 15]$, the gaze density map can then be represented by vector $X = [x_1, x_2, \ldots, x_{300}]$ with $x_i = 1$ for $i = u + (v - 1) \times 20$ and $x_i = 0$ otherwise. Finally, for each test frame, we compute feature vector F and generate the predicted map $P = F \times W$ which is then reshaped to a 20×15 saliency map. The maximum of this map is used to direct spatial attention.

k Nearest Neighbor Classifier (kNN). We also implemented a non-linear mapping from features to saccade locations. The attention map for a test frame is built from the distribution of fixations of its most similar frames in the training set. For each test frame, k most similar frames (using the Euclidean distance) were found and then the predicted map was the weighted average of the fixation locations of these frames (i.e., $X^i = \frac{1}{k}\sum_{j=1}^{k} D(F^i, F^j)^{-1} X^j$ where X^j is the fixation map of the $j - th$ most similar frame to frame i which is weighted according to its similarity to frame i in feature space (i.e., $D(F^i, F^j)^{-1}$). We chose parameter k to be 10 which resulted in good performance over train data as well as reasonable speed.

In addition to the above, we also devised two brute-force yet powerful predictors. The first one is simply the average of all saccade positions which we call **Average Fixation Map (AFM)** during the time course of a task over all m training frames (i.e., $AFM = \frac{1}{m}\sum_{j=1}^{m} X^j$). In dynamic environments used in this paper, since frames are generated on the fly and there are few fixations per frame, aligning frames (contrary to movies) is not possible. If a method could dynamically predict eye movements on a frame-by-frame basis, then achieving a higher accuracy than AFM is possible. AFM map is also the solution of the regression with a constant input, and is the output of our Bayesian

[3] Publicly available at: http://www.ics.uci.edu/~ihler/code/kde.html

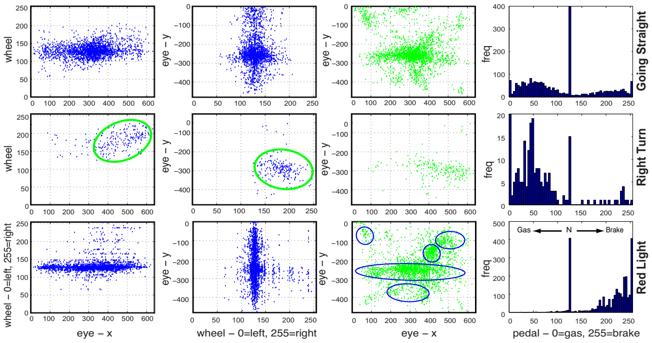

Figure 2. Correlation between actions and saccade positions. Rows indicate events (each frame was manually tagged based on its event). Columns from left to right include: $wheel$ vs. $eye - x$, $eye - y$ vs. $wheel$, saccade coordinates during the dame ($eye - x$ vs. $eye - y$), and frequency of pedal positions for DS game. Blue ellipses in the 3rd column indicate objects in the scene (see Fig. 1). Similar trends happen in the other games which eventually could help us in prediction of next saccade location.

model with one variable ($P(X)$ only). The second predictor is a **central Gaussian filter (Gauss)**. The rationale behind using this model is that humans tend to look at the center of the screen when game playing (center-bias or photographer-bias issue [31] by game design construction), therefore a central Gaussian blob may score well when datasets are centrally biased (See Figs. 3 and 6). Instead of using a fixed-size Gaussian for all games, we fitted a 2D Bivariate Gaussian to the fixation data of each game using ML algorithm:

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} exp\Big[\frac{-z}{2(1-\rho^2)}\Big] \qquad (7)$$

where $z = \frac{(x-\mu_x)^2}{\sigma_x^2} - \frac{2\rho(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y} + \frac{(y-\mu_y)^2}{\sigma_y^2}$ and ρ is the correlation coefficient between x and y (i.e., $\rho = \frac{\sum_{xy}}{\sigma_x\sigma_y}$) where $\sum_{xy}$ is the covariance matrix.

3. Quantitative Results

Here we report results of our approach for predicting saccades (jumps in eye movements to bring the relevant object/location to the fovea)[4]. While we only process those frames in which a saccade happened, our method is easily applicable for predicting fixations (one for each frame).

3.1. Eye Tracking and Data Gathering

To test our models, we have collected a large amount of multi-modal data from subjects playing video games. We intend to share our data and accompany software to encourage follow-up research on modeling top-down attention.

Human subjects in the age range 20 to 30 played 5 video games. Subjects were students of anonymous university. Some subjects played more than one game. First, in a 5 min training session, aim and rules of the game as well as buttons of playing device were explained to the subject. Subjects were then asked to play the game to become familiar with the gaming environment. After training, in a test session, subjects played a different scenario of the game than during training (e.g., a different game level) without experimenter's intervention. They had different adventures in games from each other. Before the test session started, the eye tracker (ISCAN Inc. RK-464) was calibrated using 9 point calibration scheme. Subject's head was placed on a chin-rest at the distance of $130cm$ from the screen, yielding a visual field of $43^o \times 25^o$. Subject's right eye was recorded. Along with frames and fixations, subject's actions were also logged. A computer with Windows OS ran the PC games (frame rate $30Hz$), logged actions (frequency $62Hz$), and sent frames to a computer with Linux Mandriva OS that displayed and saved frames for later analysis. Another windows machine controlled the eye tracker camera and recorded fixations ($240Hz$). All computers communicated via a LAN network and their clocks were synchronized. Each data item had a time stamp which allowed us to align frame, action, and fixation data after recording.

Stimuli. To evaluate the power of our model, we applied it to 5 games with different task algorithms and visual renderings. For some games, scenes change considerably but for some others background scene is nearly constant making gist features less variable and informative.

Two of the games are driving games. The first one, *3D Driving School (DS)* is a driving emulator with simulated traffic conditions. Players must follow the route and European traffic rules defined by the game. An instructor will tell the players where to go by a text in a semi-translucent box above the screen and/or a small arrow on the top-left corner. Players use automatic transmission to drive around the entire course. This game has only dashboard view, an inside view from the driver-side towards the road. The second driving game, *18 Wheels of Steel (WS)* is a semi/truck simulator. In this game, players control a big rig to a specific destination, to retrieve money rewards for delivering a trailer. Players must drive carefully as the truck cannot accelerate/brake suddenly due to its mass. In this game, players were told to always make a left turn since there is no explicit instruction on the screen telling where to go. Players also used first-person/bumper view. Correlations between fixation patterns and driving events were found that can help detecting driver behavior's and intention (Fig. 2). Fig. 3 shows the average fixation map for DS game and it's corre-

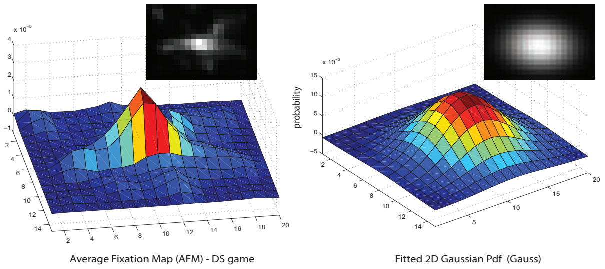

Figure 3. Average fixation map for the DS game and it's corresponding learned Gaussian map $\mu = [8.75\ 10.5]$ and $\sum = [7.85\ 0.52; 0.52\ 14.5]$.

[4] Thresholds to detect saccades were set to a velocity of $20°/s$ and an amplitude threshold of $2°/s$.

Game	# Sacc.	# Subj	Dur.	# Frames	Size	Action
DS	6382	10	10 min	180K	110	J
WS	4849	10	10 ”	180K	110	J
SM	1482	5	5 ”	45K	26	J
BS	1763	5	5 ”	45K	26	M
TG	4602	12	∼ 4.5 ”	99K	57	N/A

Table 1. Summary statistics of our data including overall number of saccades, subjects, durations per subject, frames, sizes in GB, and action types (J indicates joystick and M stands for mouse).

sponding fitted Gaussian model.

The third game, *Super Mario Bros (SM)*, is a classic 2D-side-scrolling action game. Players control Mario to a flagpole to finish the level. Mario grows bigger if it consumes a mushroom and can shoot fireballs if it consumes a flower. There are various enemies that can be killed by stomping on them or shooting fireballs. In this game, players were expected not to take any means of shortcut such as running on ceiling, teleport pipes, or warp points. Actions in this game are (x, y) position of joystick ([0, 255] for left/right, up/bottom) and status of 3 binary buttons including *Start, Jump,* and *Fire/Run*.

The fourth game called *Burger Shop (BS)* is a 2D time-management game. Under time pressure, players serve customers who order food items such as burgers and fries that must be assembled from a conveyor belt that brings ingredients. The game ends when all customers are served. For this game, actions include mouse (x, y) position as well as status of the mouse buttons (i.e., *Left, Middle,* and *Right*).

The fifth game, *Top Gun (TG)*, is a flight-combat simulator. Players control a jet-fighter plane that can lock targets and shoot missiles, use afterburners to speed up, and do air maneuvers. The main objective of the game is to completely destroy all targets on air and on the ground. Players use first-person view in this stimuli. Currently, we do not have motor actions for this game.

Table. 1 shows summary statistics of video game data.

3.2. Evaluation Metrics

To quantify how well model predictions matched observers' actual eye positions, we used two metrics:

Normalized Scanpath Saliency (NSS). NSS [34] is defined as the response value at the human eye position (x_h, y_h) in a model's predicted gaze density map that has been normalized to have zero mean and unit standard deviation: $NSS(t) = \frac{1}{\sigma_{s(x)}}\big(s(x(t)) - \mu_{s(t)}\big)$ for frame at time t. An NSS value of unity indicates the subject's eye position falls on a region whose predicted density is one standard deviation above average. Meanwhile, an NSS value of zero or lower means that the model performs no better than picking a random position on the map.

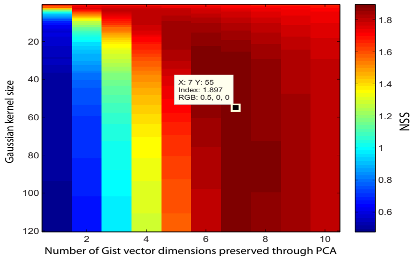

Figure 4. Grid search for best parameters (KDE kernel width and PCA dimensions of the Gist vector; Sec. 2) for DS game over train data.

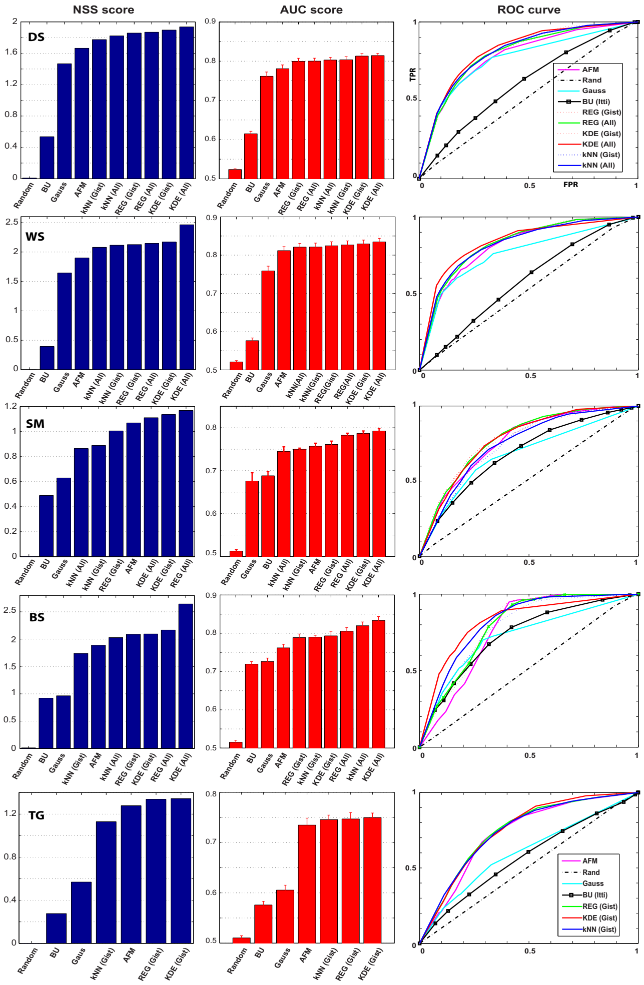

Figure 5. Prediction accuracy of our KDE model, Itti *et al.* [7], classifiers also implemented here, as well as brute-force predictors (AFM and Gaussian) for 5 video games using NSS and AUC (ROC) scores. KDE model with all features, KDE (All), results in the best performance in all cases, KDE with only Gist feature outperforms the other compared models.

Area Under the Curve (AUC). Here, a model's saliency map is treated as a binary classifier on every pixel in the image; pixels with larger saliency values than a threshold are classified as fixated while the rest of the pixels are classified as non-fixated. Human fixations are used as ground truth. By varying the threshold, the Receiver Operating Characteristic (ROC) curve is drawn as the false positive rate vs. true positive rate, and the area under this curve indicates how well the saliency map predicts actual human eye fixations [14]. Perfect prediction corresponds to a score of 1.

Results. In the first experiment, we trained the model over each separate game. Each game segment has a variable number of saccades for each subject. Training was done over saccades of $K - 1$ subjects and tested over saccades of the remaining test subject. In each training phase, the best kernel width and PCA dimensions of gist vector (see Sec. 2) were found using grid search. Fig. 4 shows

Game	ICL [17]	SDSR [20]	GBVS [24]	AIM [14]	SUN [19]	Gauss [31]	AFM	KDE(C-1)	KDE(C-2)	KDE(C-3)
DS	0.57	0.54	0.73	0.62	0.658	0.76	0.78	**0.82**	**0.82**	**0.82**
	0.19	0.05	0.948	0.54	0.30	1.47	1.66	**1.9**	**1.91**	**1.95**
WS	0.52	0.41	0.73	0.55	0.51	0.76	0.81	**0.83**	**0.83**	**0.84**
	0.27	-0.2	1.25	0.66	0.19	1.64	1.9	**2.18**	**2.21**	**2.46**
SM	0.61	0.69	0.72	0.67	0.62	0.67	0.75	**0.78**	**0.79**	**0.79**
	0.59	0.74	**1.21**	0.77	0. 33	0.62	1.07	**1.13**	**1.21**	1.11
BS	0.72	0.61	0.73	0.69	0.72	0.72	0.76	**0.79**	**0.81**	**0.84**
	1.04	0.54	1.1	0.80	1.2	0.96	1.89	**2.1**	**2.2**	**2.7**
TG	0.62	0.5	0.622	0.6	0.6	0.6	**0.73**	**0.75**	**0.75**	-
	0.58	0.01	0.55	0.51	0.29	0.57	**1.28**	**1.36**	**1.34**	-

Table 2. AUC(1st rows) and NSS scores(2nd rows) of 5 state-of-the-art models and ours over our data. Numbers in bold show best two models in each row. In almost all cases, while other models fall below Gaussian and AFM models, KDE (All) scores the best. In some cases, regression and KNN may score the best (cf. Fig. 5). C-x stands for Case x (See Sec. 2.1).

an example of best parameters over one training session of the DS game. Fig. 5 shows NSS and AUC scores, as well as ROC curves for baseline models and all variants of our model for each individual game. Over all games, KDE with all features (case 3) resulted in the best performance followed by case 2: KDE (Gist + Prev. sacc). KDE with only Gist feature outperformed classifiers with Gist, which indicates advantage of the KDE approach for using this feature compared with regression [6]. Random predictor (a random value for each location) has zero NSS and AUC near 0.5. AFM predictor achieved higher scores than BU [7] and Gaussian models over all games, indicating that eye movements were likely mostly guided top-down and BU influences were weak. AFM outperformed classifiers over the SM game, indicating that Gist is not a good predictor for this game; but when we added previous saccade position and actions, classifiers and KDE performed the best. Using action features alone in the kNN classifier resulted in NSSs of 1.41 and 1.80 for the DS and WS games, respectively higher than Gaussian and close to AFM of each game.

Table 2, shows accuracy of 5 state-of-the-art bottom-up saliency models, Gaussian, and AFM. In previous research, these models have achieved the highest scores over eye movement datasets for free-viewing task. Here, almost all of these models perform worse than AFM, while our approaches (KDE (All) and KDE (Gist)) perform higher with a large margin. This again indicates that, while bottom-up saliency models fail to account for eye fixations in our tasks which have a strong top-down component, our new models are able to capture a large amount of task-driven saccades.

In the second experiment, we trained the KDE models over one of two driving games and tested it on the other to assess the generalization power of our approach over different tasks. As Table 3 shows, training on a similar game results in higher accuracy than random, and close to performance of Gaussian and AFM predictors of each game shown in Table 2. Applying the AFM of games to each other resulted in higher accuracy than the KDE models, probably because one constant in both games is that subjects look at the center. Since actions and sequence of fixations are specific for each game, adding them slightly drops the performance (KDE (Gist) vs. KDE (All)).

Train on	DS	WS
Test on	WS	DS
AFM	0.80 (1.74)	0.75 (1.51)
KDE (Gist)	0.80 (1.64)	0.74 (1.40)
KDE (All)	0.79 (1.62)	0.73 (1.51)

Table 3. Confusion matrix of training models on one driving game and applying it to the other one using AUC and NSS(parenthesis).

	Gist [10]		**HOG [30]**	
Game	kNN	REG	kNN	REG
DS	0.80 (1.77)	0.8 (1.86)	0.81 (1.88)	**0.81 (2.05)**
SM	0.75 (0.88)	0.76 (1.01)	0.74 (0.97)	**0.79 (1.23)**

Table 4. Comparing AUC and NSS (in parenthesis) of Gist model of Siagian *et al.* [10] and HOG features for saccade prediction using kNN and regression classifiers for 3D Driving School and Super Mario games. Dimensionality of Gist vector is 714 and dimensionality of HOG is 4800. Only for REG (HOG) dimensionality of HOG is reduced to 95% of its variance which preserved about 900 D for DS and 500 for Mario game.

In the third experiment, we aimed to compare the power of HOG features [30] and the Gist features of Siagian *et al.* [10]. The notion behind using HOG features is that they encode rich structural information from the entire scene and have been very successful in object detection. Table 4 shows the performance of kNN and regression classifiers over DS and SM games. HOG features were better descriptors of the scene and conveyed more information regarding saccade locations over both games and using both classifiers. However, because calculating 8 orientation channels in HOG makes it slower than gist in [10] (about 2 times) which uses 4, here we performed experiments using the second one. HOG also generates high dimensional feature vectors which makes it hard to store and work with.

Figure 6 shows sample frames of video games with corresponding saliency maps from models. Predicted maps by our models show dense activity at task relevant locations thereby narrowing attention and leading to higher NSS and AUC scores. These maps change per frame as opposed to the static AFM and Gaussian models.

4. Discussion and Future Work

We proposed a unified Bayesian approach that is applicable to a large class of everyday tasks where global scene knowledge, the sequence of fixated locations, and actions, constrain future eye fixations. In addition to the above-mentioned factors, there might be other general features influencing task-driven attention. Our framework allows easy incorporation of those features for saccade prediction.

An important application of our model is quantitative analysis of differences among populations of subjects (e.g., young vs. elderly or novices vs. experts) in complex tasks such as driving. It can also be useful for assistant technologies for demanding tasks, human computer interaction, context aware systems, and health care.

Although employed features convey information regarding the next saccade, it is still possible to gain higher performance by knowing more about the scene. For instance, by

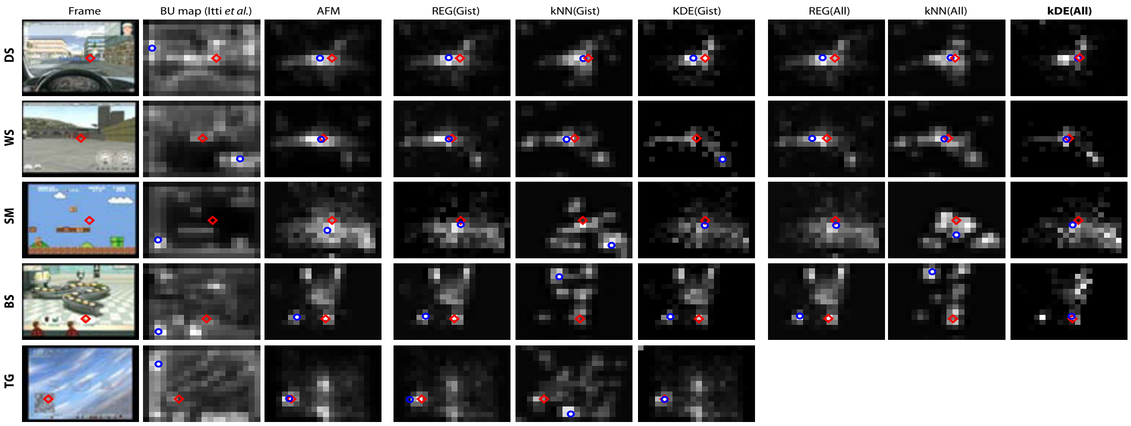

Figure 6. Sample frames of video games and corresponding predicted maps of models. Red diamond indicates the human fixation and blue circles is the maximum point of each map. Smaller distance hence means better prediction. Currently we don't have action data for TG game.

calculating the number or state of task-related objects. Such approach, however, has the drawback that for each task, relevant variables and interactions among them should be defined, thus limiting its generalization. We are now investigating the role of local context ($P(O|L_v)$ in (1)) in modulation of top-down attention. Instead of predicting fixation locations, it may be more efficient to bias the visual system toward features of a relevant object within a global context. While the exact fixated location at nearly the same gist may change based on recent history of saccades and actions, looking for a given object rather than a given location may exhibit stronger invariance. Also, extraction and addition of subjective factors such as fatigue, preference, and experience into our model would be an interesting next step.

Supported by the National Science Foundation (grant number BCS-0827764), and the Army Research Office (W911NF-08-1-0360 and W911NF-11-1-0046), and U.S. Army (W81XWH-10-2-0076). The authors affirm that the views expressed herein are solely their own, and do not represent the views of the United States government or any agency thereof.

References

[1] A. Yarbus. Eye movements during perception of complex objects. L. Riggs, editor, Eye Movements and Vision, 1967. 2

[2] M. Land and M. Hayhoe. In what ways do eye movements contribute to everyday activities? *Vision Research*, 2001. 2

[3] M. Hayhoe and D. Ballard. Eye movements in natural behavior. *Trends in Cog. Sci.*, 9(4):188-193. 2005. 2

[4] D. Ballard, M. Hayhoe, and J. Pelz. Memory representations in natural tasks. *J. of Cog. Neurosci.*, 7(1):66-80, 1995. 2, 3

[5] M.F. Land and D.N. Lee. Where we look when we steer. *Nature*, 1994. 2

[6] R.J. Peters and L. Itti. Beyond bottom-up: Incorporating task-dependent influences into a computational model of spatial attention. *CVPR*, 2007. 2, 4, 7

[7] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *PAMI*, 1998. 1, 2, 3, 6, 7

[8] V. Navalpakkam and L. Itti. Modeling the influence of task on attention. *Vision Research*, 45(2): 205-231, 2005. 2

[9] N. Sprague and D. Ballard. Eye Movements for reward maximization. *NIPS*, 2003. 2

[10] C. Siagian and L. Itti, Rapid biologically-inspired scene classification using features shared with visual attention. *PAMI*, 29(2):300-312, 2007. 2, 3, 7

[11] K. Ehinger, B. Hidalgo-Sotelo, A. Torralba, and A. Oliva. Modeling search for people in 900 scenes: a combined source model of eye guidance. *Visual Cognition*, 2009. 1, 2

[12] A.M. Treisman and G. Gelade. A feature integration theory of attention. *Cognitive Psych.*, 12:97-136, 1980. 1

[13] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 1985. 1

[14] N.D.B. Bruce and J.K. Tsotsos. Saliency based on information maximization. *NIPS*, 2005. 1, 2, 6, 7

[15] D. Gao, S. Han, and N. Vasconcelos. Discriminant saliency, the detection of suspicious coincidences and applications to visual recognition. *IEEE PAMI*, 2009. 1

[16] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. *CVPR*, 2007. 1

[17] X. Hou and L. Zhang. Dynamic visual attention: Searching for coding length increments. *NIPS*, 2008. 1, 7

[18] A. Torralba, A. Oliva, M. Castelhano and J. M. Henderson. Contextual guidance of attention in natural scenes: The role of global features on object search. *Psychological Review*, 2006. 1, 2, 3

[19] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, SUN: A Bayesian framework for saliency using natural statistics. *J. Vis.*, 2008. 1, 2, 7

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

[21] T. Avraham, M. Lindenbaum. ESaliency: Meaningful attention using stochastic image modeling. *PAMI*, 2010. 1

[22] V. Navalpakkam and L. Itti. An integrated model of top-down and bottom-up attention for optimizing detection speed.*CVPR*,2006. 1, 2

[23] S. Frintrop. VOCUS: A visual attention system for object detection and goal-directed search. Springer 2006. 2

[24] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. *NIPS*, 2006. 7

[25] T. Foulsham and G. Underwood. What can saliency models predict about eye movements? spatial and sequential aspects of fixations during encoding and recognition. *J. of Vision*, 8:1-17, 2008. 1

[26] W. Einhauser, U. Rutishauser, and C. Koch. Task-demands can immediately reverse the effects of sensory-driven saliency in complex visual stimuli. *J. of Vision*, 2008. 1

[27] J. Henderson. Regarding scenes. *Current directions in psychological science*, 16:219-227, 2007. 1

[28] F. Li, R. VanRullen, C. Koch, and P. Perona. Rapid natural scene categorization in the near absence of attention. *PNAS*, 2002. 1

[29] S. Ullman. Visual routines. *Cognition*, 18:97-157, 1984. 2

[30] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR*, 2005. 3, 7

[31] B.W. Tatler. The central fixation bias in scene viewing: selecting an optimal viewing position independently of motor bases and image feature distributions. *J. of Vision*. 14(7):1-17, 2007. 5, 7

[32] J. Hays, A.A. Efros. Scene completion using millions of photographs. *SIGGRAPH*, 2007. 2

[33] L. Renniger and J. Malik. When is scene identication just texture recognition? *Vision Research*, 2004. 3

[34] R. Peters, A. Iyer, L. Itti, and C. Koch. Components of bottom-up gaze allocation in natural images. *Vision Res.*, 45, 2005. 6

[35] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? *CVPR*, 2010. 1

[36] S. Vijayanarasimhan and A. Kapoor. Visual recognition and detection under bounded computational resources. *CVPR*, 2010. 1

[37] H.W. Kang, Y. Matsushita, X. Tang, and X.Q. Chen. Space-time video montage. *CVPR*, 2006. 1

[38] L. Wolf, M. Guttmann, and D. Cohen-Or. Content-driven video retargeting. *ICCV*, 2007. 1