HMM-based Convolutional LSTM for Visual Scanpath Prediction

Ashish Verma, Debashis Sen
Department of Electronics and Electrical Communication Engineering, IIT Kharagpur, India
ashish.verma@iitkgp.ac.in, dsen@ece.iitkgp.ac.in

***Abstract*—The human visual system performs a dynamic process of scanning the scene by rapid eye movements and fixations, yielding a visual scanpath. We propose an approach to generate artificial visual scanpaths for natural images. A convolutional long short term memory (LSTM) neural network is employed, which learns the mapping of image features to eye fixations by modeling the sequential dependencies of the fixations in a scanpath. A novel approach of hidden Markov model (HMM) based data augmentation is presented that increases the number of available image-specific input-output pairs to train the LSTM appropriately. Both the HMM and the LSTM are designed to be consistent with existing knowledge on saccadic eye movements. Experimental results on a standard eye-tracking dataset demonstrate that the proposed approach does better than the state-of-the-art and generates realistic visual scanpath data.**

***Index Terms*—Visual scanpath prediction, eye tracking, fixations, saccades, Convolutional LSTM**

I. Introduction

Eye movements are very important attributes of the human visual system (HVS) for the perception of a scene in many tasks like driving or looking at a picture. One of the main aspects involved in visual perception of humans is a dynamic process where the eye moves to scan the scene. The human visual system has high visual acuity in a small region, the fovea, and hence our eyes process only a small central region in detail and the resolution drops rapidly towards the periphery. So to build a detailed representation of a scene, the HVS performs a dynamic process of scanning the scene by rapid eye movement called saccades. To gather more visual information HVS makes a series of such saccadic eye movements between fixation points which results in a human visual scanpath.

In computer vision, a field related to visual scanpath explored extensively is visual saliency prediction. A saliency map for an image/frame is predicted that signifies the importance of the entities present [1], [2], and is directly related to human fixation density. However, only a few approaches have been proposed to predict visual scanpaths. In one of the earliest work [3] involving saliency-based visual attention computation, Itti *et al.* modeled the visual scanpath as dynamical shifts of the focus of attention by employing winner-take-all (WTA) and inhibition of return (IoR) policies on the predicted saliency map. To predict eye movement trajectories Komogorestev *et al.* [4] introduced a mathematical model of the human eye in the form of Kalman filter which uses anatomical properties of HVS. Wang *et al.* [5] proposed a computational model to simulate saccadic scanpath on natural images. The model considered three factors namely reference sensory responses, visual working memory, which guide the eye movements and fovea-periphery resolution discrepancy. One of the earliest learning-based method for estimating human scanpaths is proposed by Liu *et al.* [6]. They also considered three different factors, namely low-level feature saliency, semantic content, and spatial position with Levy flight model to account for spatial position and hidden Markov modeling (HMM) to learn the influence of semantic content. In [7], Sun *et al.* presented a statistical framework which models the saccadic behavior and visual saliency based on super-Gaussian component (SGC) analysis. Wang *et al.* [8] proposed a bio-inspired method for fixation and scanpath prediction in which a probability map based on the foveated image saliency is used with two factors namely the saccadic biases of gaze shifts and the IoR policy. Another learning based method proposed by Jiang *et al.* [9] used least-squares policy iteration (LSPI) to a visual exploration policy from the recorded human eye tracking data. The method allows the integration of low-level and high-level cues in the learning process by considering various stages of visual exploration. In recent work [10], Xia *et al.* proposed an iterative scene representation learning framework using deep autoencoder in which the saccade is considered as an iterative process of finding the most uncertain area in the scene. Another learning framework [11] is presented by Ngo *et al.* to model the sequences with a recurrent neural network (RNN) using localized features from a pretrained convolutional neural network (CNN). They trained the model on free-viewing eye tracking data by maximizing the likelihood of a fixation sequence given an image.

In this paper, we propose a convolutional LSTM based human visual scanpath predictor. We use long short term memory (LSTM) neural network to predict fixation sequences based on the image features extracted by a pretrained CNN. The proposed method is based on existing knowledge about saccadic eye movements during free-viewing, and has advantages over the existing methods in terms of model complexity and performance. Due to the end-to-end training, our method does not assume any prior knowledge about the data, and feature extraction through CNN in a single pass makes it simpler by eliminating feature set collection and selection process. Most importantly, unlike previous works, we employ a novel data augmentation procedure by increasing the number of available training pairs through image-specific fixation sequence modeling. For this, we consider a hidden Markov

978-9-0827-9703-9/19/$31.00

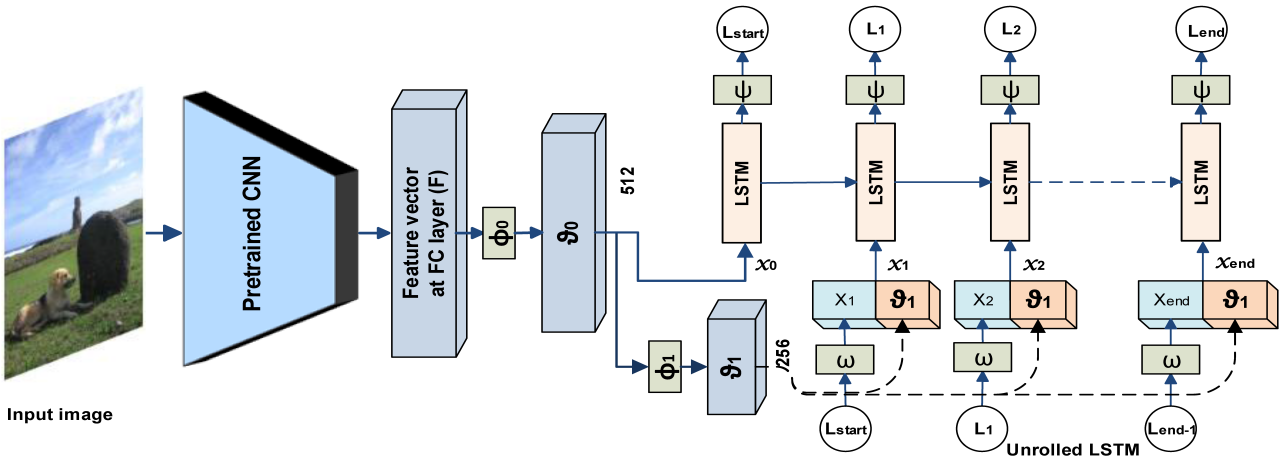

Fig. 1. Convolutional LSTM based visual scanpath Predictor trained on human and HMM generated visual scanpaths. A fixation point location prediction happens based on previous fixation point and abstract feature of the image.

model (HMM) based sequence generation, in coherence with existing knowledge on saccadic eye movements. Quantitative and qualitative experimental results are given to demonstrate the effectiveness of our proposed method.

II. Proposed Approach

The proposed HMM-based convolutional LSTM visual scanpath prediction is illustrated in Fig.1. From a given image the pretrained CNN extracts features, which is used by the LSTM network to predict the visual scanpath by learning the mapping of image features to fixation locations along with their dependencies on previous fixation locations. Since training credible LSTM network requires a large number of training samples (known input-output pair), we use HMM to generate synthetic image-specific fixation sequences.

A. Data augmentation using HMM

It is well accepted [12] that during saccadic eye movement, the fixation at a location depends on the content at that location and the previous fixation location. This dependence of the current fixation location on the previous is in agreement with Markovian property involved in HMM-based sequence modeling. Hence, we use HMM to learn a model of the fixation sequences in an image and then use the learned HMM to generate more such fixation sequences for that image. This results in an increase in the number of input-output pairs available to train our LSTM network, which is designed to predict fixation locations in any arbitrary image based on image content and previous fixation location (See Section II-D). This HMM-based increase in training pairs can be envisioned as a data augmentation process, which boosts the network's invariance to discrepancies in image features at different locations present at the same sample t of multiple fixation sequences.

Further, modeling an image-specific sequence of eye fixations using HMM allows its hidden states to represent the latent image region selection criterion, while the observable outputs are image locations. An HMM with N hidden states $(S_1, S_2, S_3..., S_N)$ is comprised of three parameters $\lambda = (\pi, \Theta, \Phi)$, where $\pi \in \mathbb{R}^N$ is a vector representing the prior probability distribution of the hidden states, $\Theta \in \mathbb{R}^{N\times N}$ is the transition matrix consisting of probabilities of transition from one state to other, $\Phi \in \mathbb{R}^{N\times K}$ is the emission matrix with probability values of observations given a certain state with K denoting the number of emissions.

The HMM model for an image can be trained by using multiple human visual scanpaths in that image, represented by $\{O_n \mid n = 1, 2..M\}$, where $O_n = \{{o_1}^{(n)}, ..., {o_{T_n}}^{(n)}\}$ is the n^{th} training sample (i.e., a human visual scanpath from one subject in an image), and M is the total number of subjects in the image, T_n is the length of a scanpath of the n^{th} subject. First, we initiate a model as $\lambda = (\pi, \Theta, \Phi)$ for an image and then use that to calculate the re-estimation of the HMM parameters by employing Baum-Welch [13] method to improve the probability of an O_n being observed from the model. So we trained individual HMM for an image based on the available visual scanpaths on that image and then used the model to generate artificial scanpaths for that image.

B. Feature extraction through CNN

The recent works [14] [15] showed that Convolutional Neural Network (CNN) is capable of learning the features which represent the semantics in the images for object detection. Moreover, many works have shown that pretrained CNN on ImageNet dataset [16] for image classification can be used as fixed feature extractor by removing the later fully-connected layers for many tasks like object classification, detection, and segmentation. We have used the filter responses of the second last layer of the 152-layer ResNet of [17] as the image feature set, which we represent as $\{F\}$. The feature set $\{F\}$ of an image is converted into another feature set $\{\vartheta_0\}$ with reduced dimension as required by our LSTM network using a fully-connected linear layer ϕ_0. We feed this feature set $\{\vartheta_0\}$ of an image to LSTM network at the first time instance. Then another fully-connected layer ϕ_1 is used on ϑ_0 to produce

abstract image feature ϑ_1, which is fed into the LSTM at the later time instances along with the fixation locations (ground truth in training phase and previous predicted in testing phase). Representation X_t of fixation locations obtained by passing them through a fully-connected linear layer ω are concatenated with ϑ_1.

C. Spatial quantization of image

As fixation location in the visual scanpath represents the screen location where the point-of-gaze rests within a range of visual angle for at least a fixed period [18], the fixation location represents a small region of an image. So first, we resize all the images into 512×512 pixels and then divide the whole image into 1024 small regions $\{L_i\}$ each of size 16x16 pixels. Then, all the fixation locations are binned into one of the small regions and represented by the center of the region. Hence, we can represent the visual scanpath as a sequence of these small regions corresponding to its fixation locations. Each eye-fixation regions in such a sequence correspond to an entry in our dictionary set γ, which also contain a start and end token for the sequence. Then each of the element in the dictionary is embedded into a 256-dimensional feature vector by a single layer perceptron ω which is learned with the training of the whole model.

D. Convolutional LSTM based Visual Scanpath Predictor

The human eye makes a sequence of saccades between fixations to perceive an image. The first eye fixation on the image depends on the features in whole image, whereas the next fixations are driven by the image features and the previous fixation [12]. We have designed our approach based on this fact, which we discuss here in detail.

Long Short-term Memory (LSTM) [19] is one of the most popular deep learning algorithm which learns the sequential information in the data. LSTM is developed to solve the vanishing and exploding gradient problem of RNN [20] by including memory blocks in its recurrent connections. For a given sequence input as $x_t, t \in T$ (where T is the length of sequence), an LSTM unit recursively calculates the output o_t and hidden state h_t at each time step until T by following:

$$\begin{cases} i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + b_i\right), \\ f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right), \\ o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + b_o\right), \\ g_t = W_{xg}x_t + W_{hg}h_{t-1} + b_g, \\ c_t = f_t \odot c_{t-1} + i_t \odot \phi\left(g_t\right) \\ h_t = o_t \odot \phi\left(c_t\right) \end{cases} \tag{1}$$

where x_t and h_t are the input and hidden states for the LSTM unit at the t^{th} time step, i_t, f_t, o_t, g_t, c_t represents the input gate, forget gate, output gate, gate gate, and memory cell respectively. σ is the sigmoid function, and ϕ is the hyperbolic tangent function.

The features of an image extracted by the CNN as discussed in section II-B, are represented by $\{\vartheta_0\}$ and fed into LSTM at first time instant for the visual scanpath prediction on the image. As the hidden state $h_t \in \mathbb{R}^n$ in LSTM varies over time t, the t^{th} fixation location (L_t) in the visual scanpath is selected from the dictionary γ according to the probability distribution over dictionary $\left(p_t \in \mathbb{R}^{|\gamma|}\right)$ generated by a fully connected layer with softmax non-linearity connected to the hidden state h_t of LSTM. The feature vector X_{t+1} of next fixation location L_{t+1} concatenated with more abstract image feature ϑ_1 will be given as an input $x_{t+1} \in \mathbb{R}^{|m|}$ into LSTM network in the next time step, which results in the transition of state from h_t to h_{t+1}. The visual information ϑ_0 and ϑ_1 of image act as the source which guides the LSTM in generating p_t and hence L_t, based on input and output models ω and ψ.

These process flow of convolutional LSTM based visual scanpath predictor is governed by the following equations:

$$x_0 = \vartheta_0 = \phi_0\left(F\right) = W_{F\vartheta_0}F + b_F \tag{2}$$

$$\vartheta_1 = \phi_1\left(\vartheta_0\right) = W_{\vartheta_0\vartheta_1}\vartheta_0 + b_{\vartheta_1} \tag{3}$$

$$h_t = LSTM\left(h_{t-1}, x_t \oplus \vartheta_1\right) \quad t > 0 \tag{4}$$

$$L_t \sim p_t = \psi\left(h_t\right) \tag{5}$$

$$X_t = \omega\left(W_{Lx}L_{t-1} + b_L\right), \quad t > 0 \tag{6}$$

where ϕ_0 used in the (2) is linear model to reduce the dimension of feature vector such that it can be given to the LSTM, ω used in (6) is a linear embedding model which embeds the fixation locations, ψ used in the (5) is a fully connected layer with softmax non-linearity which converts hidden states into probability distributions over the dictionary γ. The symbol $\oplus$ represents concatenation operation, and ϕ_1 reduces the dimension of image feature ϑ_0 such that after concatenated with embedded feature vector X_t the resultant vector will have proper dimension as input to LSTM. The generation of h_t from h_{t-1} and x_t using LSTM as used in (4) follows the expressions given in the (1). The processes described in (4) to (6) are recursively applied with every time step to get the sequence of locations for the input image.

An argmax decoding is applied to recover the fixation location sequence Ł for a given image from the conditional probabilities, and can be represented as:

$$L_{t+1} = \underset{j \in \gamma}{argmax} P\left(L_{t+1} = j \mid I, L_1, L_2 ... L_t\right) \tag{7}$$

So for a given image I the Convolutional LSTM based human visual scanpath predictor generates a sequence of fixation locations $\{L_1, L_2 ... L_T\}$,$L_t \in \gamma$. We train the Convolutional LSTM based visual scanpath predictor on an image from human scanpaths (15%) and HMM generated scanpaths (85%) on the image. The ground truth sequence of fixation locations $\{\hat{L}_1, \hat{L}_2 ... \hat{L}_T\}$,$\hat{L}_t \in \gamma$ is a ground truth visual scanpath.

For each image, the loss function used is categorical cross entropy loss over fixation locations between ground truth and predicted visual scanpath, which is defined as:

$$\pounds = -\sum_{t=1}^{T} \log p\left(L_t = \hat{L}_t \mid I, \hat{L}_{t-1}, ... \hat{L}_1\right) \tag{8}$$

where $\hat{L}$ denotes the ground truth terms. During the training of the model the input terms $\{\hat{L}_{t-1}, \hat{L}_{t-2} ... \hat{L}_1\}$ are the ground truth fixation locations.

III. Experiments

A. Dataset

We have used MIT dataset [21], a publicly available dataset for training and evaluation of our method as considered in [9] and [11]. MIT dataset is recorded human eye tracking data in a free viewing setting. The dataset contains eye tracking data from 15 people across 1003 natural images.

B. Evaluation Metric

For the evaluation of predicted visual scanpath, we used sequence sccore (SS) as an evaluation metric proposed in [22]. This method first uses mean-shift clustering to cluster all the human fixation points in a given image considering an optimal bandwidth parameter that maximizes the interaction rate among the clusters as given in [9]. Then each of these clusters with the corresponding fixations is represented by a unique character yielding a string representation of a sequence of fixations. As in [9], the Needleman-Wunsch string matching is then used to determine the similarity score between fixation sequences. For a predicted scanpath in an image, the similarity scores are calculated against all human scanpaths for the image, whose average gives the final evaluation score. The overall interobserver similarity for an image is calculated by averaging all the scanpath similarity scores obtained between every two human scanpaths for that image.

C. Training Details

HMM: we set the number of hidden states $N = 10$. We assume the continuous emission probability distribution to be Gaussian. The values of the transition matrix and prior distribution are initialized from uniform distributions. The HMM for an image is trained for all the available human visual scanpaths of different lengths on that image and then used to generate artificial visual scanpaths comprising of 6 fixation points.

ConvLSTM: In the convolutional LSTM based visual scanpath predictor, the image feature set $\{F\}$ described in Section II-B has 204800 components. We have used a single layer LSTM with 512 hidden units. Therefore, both $\{\vartheta_0\}$ and $\{\vartheta_1\}$ concatenated with representation of fixation location X_t, elaborated in Section II-B, are 512-dimensional feature vectors. We trained the model with dropout and parameter optimization using Adam optimizer with a learning rate of 0.00001 and exponential decay rate of values 0.9 and 0.999 for the first and second moment, respectively.

D. Results and Discussion

The recent studies [9], [11] have shown that Judd's saliency [21] method's performance with IoR and WTA for visual scanpath prediction is better than other saliency based methods, and also is comparable to the proposals in [9], [11], the state-of-the-art. We compare our approach with several saliency based methods [3], [23], including Judd's [21]. As shown in Fig. 2, our method performs substantially better than above said methods, where the sequence scores of our method are obtained by 10-fold cross-validation, and in a similar vein, the scores for the other methods are averages over all images. Since the codes for the state-of-the-art methods [9], [11] have not been made available, so in Fig. 3, we compare our approach with the state-of-the-art in an unbiased manner by evaluating the improvement achieved over Judd's method as the baseline. We consider Judd's method, as it's performance has been reported both in [9] (LSPI) and [11] (RNN) in their corresponding settings. From Fig. 3, we observe the performance improvement of three methods (LSPI, RNN and ours) over the corresponding baseline Judd's method. The amount of improvements is also quantified in Table I. From both Fig. 3 and Table I, we see that the proposed approach achieves the most improvement over Judd's. Note that inter-observer performances cannot be used for such an unbiased comparison as they are not affected the evaluation settings. Human and predicted visual scanpaths by our and other baseline models on a few images are shown in Fig. 4, which demonstrates the effectiveness of our approach.

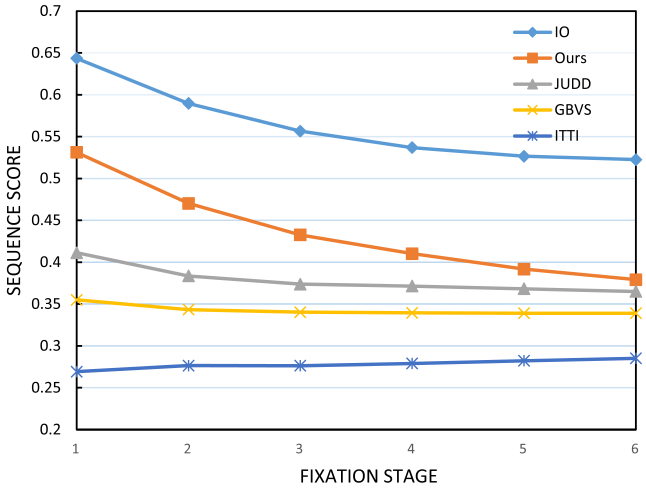

Fig. 2. Evaluation of our model (orange square) and baseline models Judd [21] (gray triangle), GBVS [23] (yellow cross), Itti [3] (blue star), and inter-observer performance (blue diamond) on the MIT dataset.

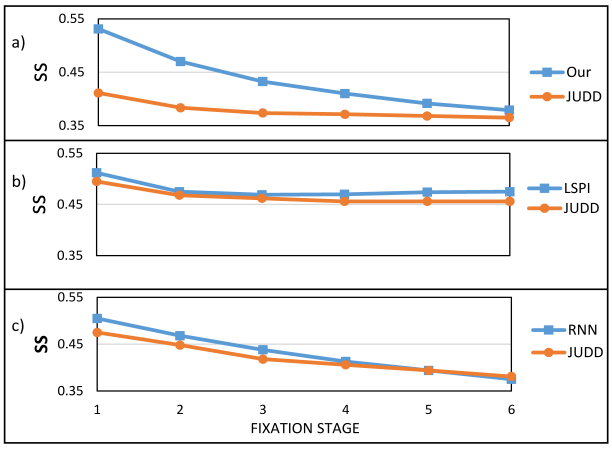

Fig. 3. Sequence Score (SS) plot for a) our model, b) LSPI [9], and c) RNN [11] with corresponding Judd's method [21].

TABLE I
SUM OF DIFFERENCE OF SEQUENCE SCORE BETWEEN METHOD AND BASELINE JUDD'S METHOD

Method	Difference of sequence score (SS) between method and baseline Judd's saliency method						Sum of Diff
Fixation stage	1	2	3	4	5	6	
LSPI	0.017	0.12	0.007	0.014	0.018	0.019	0.195
RNN	0.03	0.02	0.02	0.007	0	-0.006	0.071
Our Method	0.119	0.086	0.059	0.038	0.023	0.014	0.339

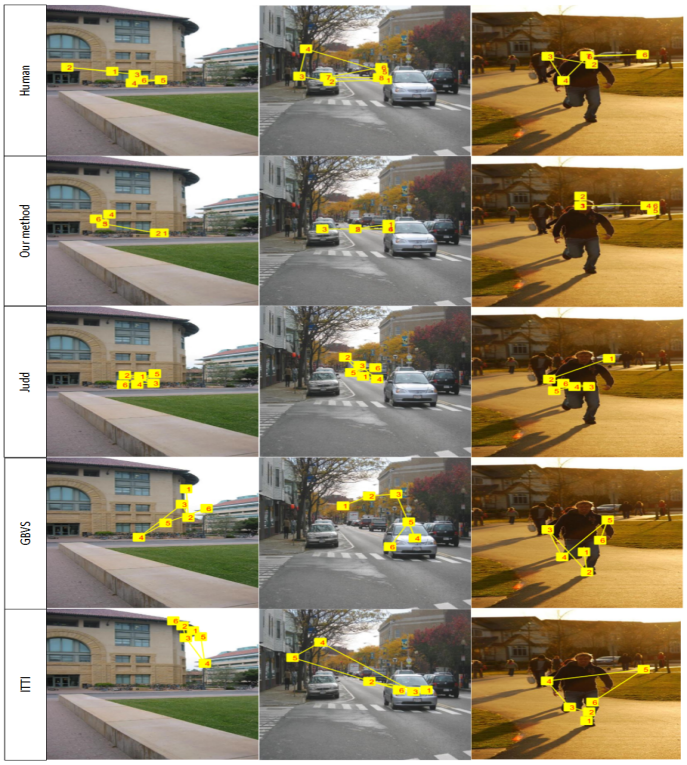

Fig. 4. Qualitative evaluation of our approach and baseline methods on MIT dataset, along with Human ground truth.

IV. CONCLUSION

A novel method for artificial visual scanpath generation on images is discussed. The generation is achieved using a convolutional long short term memory neural network that contains a pretrained CNN. The LSTM is trained on available human visual scanpaths and, HMM generated image-specific visual scanpaths to learn a mapping between image features and fixation locations given the previous fixation location. Our approach, when evaluated on a standard dataset, has been found to produce results quantitatively better than the state-of-the-art. Qualitative analysis has also shown that meaningful visual scanpaths are generated by the proposed method.

REFERENCES

[1] E. Vig, M. Dorr, and D. Cox, "Large-scale optimization of hierarchical features for saliency prediction in natural images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2798–2805.

[2] N. Bruce and J. Tsotsos, "Saliency based on information maximization," in *Advances in neural information processing systems*, 2006, pp. 155–162.

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

[4] O. V. Komogortsev and J. I. Khan, "Eye movement prediction by oculomotor plant Kalman filter with brainstem control," *Journal of Control Theory and Applications*, vol. 7, no. 1, pp. 14–22, 2009.

[5] W. Wang, C. Chen, Y. Wang, T. Jiang, F. Fang, and Y. Yao, "Simulating human saccadic scanpaths on natural images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 441–448.

[6] H. Liu, D. Xu, Q. Huang, W. Li, M. Xu, and S. Lin, "Semantically-based human scanpath estimation with hmms," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3232–3239.

[7] X. Sun, H. Yao, R. Ji, and X.-M. Liu, "Toward statistical modeling of saccadic eye-movement and visual saliency," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4649–4662, 2014.

[8] Y. Wang, B. Wang, X. Wu, and L. Zhang, "Scanpath estimation based on foveated image saliency," *Cognitive processing*, vol. 18, no. 1, pp. 87–95, 2017.

[9] M. Jiang, X. Boix, G. Roig, J. Xu, L. Van Gool, and Q. Zhao, "Learning to predict sequences of human visual fixations," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 6, pp. 1241–1252, 2016.

[10] C. Xia, F. Qi, and G. Shi, "An iterative representation learning framework to predict the sequence of eye fixations," in *IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2017, pp. 1530–1535.

[11] T. Ngo and B. Manjunath, "Saccade gaze prediction using a recurrent neural network," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 3435–3439.

[12] C. Collins and G. R. Barnes, "Predicting the unpredictable: weighted averaging of past stimulus timing facilitates ocular pursuit of randomly timed stimuli," *Journal of Neuroscience*, vol. 29, no. 42, pp. 13 302–13 314, 2009.

[13] J. Baker, "The dragon system–an overview," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 23, no. 1, pp. 24–29, 1975.

[14] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware cnn model," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1134–1142.

[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 1137–1149, 2017.

[16] J. Deng, R. Socher, L. Fei-Fei, W. Dong, K. Li, and L.-J. Li, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, vol. 00, 06 2009, pp. 248–255.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[18] S. Ramanathan, H. Katti, N. Sebe, M. Kankanhalli, and T.-S. Chua, "An eye fixation database for saliency detection in images," *Computer Vision–ECCV 2010*, pp. 30–43, 2010.

[19] X. Li and X. Wu, "Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4520–4524.

[20] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[21] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *IEEE 12th international conference on Computer Vision*. IEEE, 2009, pp. 2106–2113.

[22] A. Borji, H. R. Tavakoli, D. N. Sihite, and L. Itti, "Analysis of scores, datasets, and models in visual saliency prediction," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 921–928.

[23] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Advances in neural information processing systems*, 2007, pp. 545–552.