Improving Driver Gaze Prediction With Reinforced Attention

Kai Lv, Hao Sheng, *Member, IEEE*, Zhang Xiong, Wei Li, *Member, IEEE*, and Liang Zheng, *Member, IEEE*

***Abstract*—We consider the task of driver gaze prediction: estimating where the location of the focus of a driver should be, based on a raw video of the outside environment. In practice, we output a probability map that gives the normalized probability of each point in a given scene being the object of the driver attention. Most existing methods (*i.e.*, *Coarse-to-Fine* and *Multi-branch*) take an image or a video as input and directly output the fixation map. While successful, these methods can often produce highly scattered predictions, rendering them unreliable for real-world usage. Motivated by this observation, we propose the reinforced attention (RA) model as a regulatory mechanism to increase prediction density. Our method is built directly on top of existing methods, making it complementary to current approaches. Specifically, we first use *Multi-branch* to obtain an initial fixation map. Then, RA is trained using deep reinforcement learning to learn a location prediction policy, producing a reinforced attention. Finally, in order to obtain the final gaze prediction result, we combine the fixation map and the reinforced attention by a mask-guided multiplication. Experimental results show that our framework improves the accuracy of gaze prediction, and provides state-of-the-art performance on the DR(eye)VE dataset.**

***Index Terms*—Gaze prediction, driver attention, reinforcement learning, video processing, deep learning.**

I. Introduction

AUTONOMOUS and assisted driving are some of the most active research areas in computer vision. Typically, these works have focused on lane change assistance [1], traffic signs recognition [2], and many more [3]. Recently, several works [4], [5] have advocated for a new assisted driving paradigm - driver gaze prediction. The goal of gaze prediction is to provide useful suggestions to the driver where they should focus. In this task, the gaze points are gathered from real driving scene, and are defined as the ground truth of the training dataset. In practice, gaze is defined as a probability map where each point in a given scene has a value. This value denotes how much probability this point is the gaze of the driver.

Some previous works take saliency detection methods to address the challenge of driver gaze prediction. Some methods attempt to capture salient objects or events that occur naturally in the environment as driver gaze [6]–[8]. Other models combine saliency details with motion cues to handle gaze prediction task [4], [9]–[11]. These methods do not directly use ground truth human gaze and instead attempt to approximate the task purely via environment cues. Recent works [5], [9] propose to specifically use supervised gaze prediction to achieve higher performance. For example, Palazzi *et al.* [5] propose a model based on C3D [12] that takes videos of driver outside environment as input, allowing the model to explicitly take into account the temporal dimension.

Multi-branch [9] is another work solving driver gaze prediction in a supervised strategy and is considered state-of-the-art. Multi-branch has the architecture of three branches: RGB image, optical flow and semantic segmentation. Each branch provides complementary details for the overall model to contribute to the final prediction.

However, as shown in Fig. 1(c), the Multi-branch [9] tends to produce scattered gaze prediction maps. The prediction map of Multi-branch does not reflect the distribution of the ground truth gaze, as shown in Fig. 1(b), where gaze tends to appear in tightly clustered areas. In other words, the predicted gaze should mainly be concentrated in one area. The concentrated map reflects the nature of human perception: the focus of driver gaze is localized to the most important part of the environment at any given point in time.

In contrast to most previous approaches, our work aims to introduce attention as a regulatory mechanism to increase prediction density and accuracy. In this paper, the attention means the location where the driver should mainly focus on. We believe that there is only one attention at a moment. The overall model has the following advantage. By introducing attention, the proposed method can output more accurate and reasonable

Manuscript received May 13, 2020; revised October 18, 2020; accepted November 3, 2020. Date of publication November 17, 2020; date of current version November 18, 2021. This work was supported in part by the National Key R&D Program of China under Grant 2019YFB2101600, in part by the National Natural Science Foundation of China under Grants 61861166002, 61872025, and 61635002, in part by the Science and Technology Development Fund, Macau SAR (File no.0001/2018/AFJ), and in part by the Fundamental Research Funds for the Central Universities and the Open Fund of the State Key Laboratory of Software Development Environment under Grant SKLSDE2019ZX-04. The work of L. Zheng was supported by the Australian Research Council Discovery Early Career Award (DE200101283) funded by the Australian Government. *(Corresponding author: Hao Sheng.)*

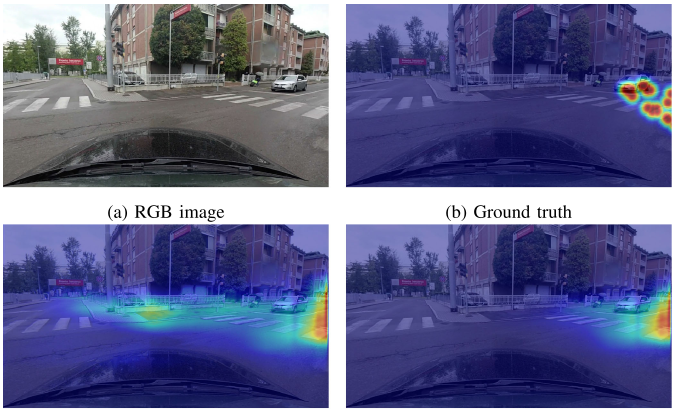
Kai Lv and Hao Sheng are with the State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University, Beijing 100191, P.R. China, with the Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China, and also with the Beihang Hangzhou Institute for Innovation at Yuhang, Beihang University, Hangzhou 311121, P.R. China (e-mail: lvkai@buaa.edu.cn; shenghao@buaa.edu.cn).

Zhang Xiong and Wei Li are with the School of Computer Science and Engineering, Beihang University, Beijing 100191, China (e-mail: xiongz@buaa.edu.cn; liwei@nlsde.buaa.edu.cn).

Liang Zheng is with the Research School of Computer Science, Australian National University, Canberra 2601, Australia (e-mail: liang.zheng@anu.edu.au).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMM.2020.3038311.

Digital Object Identifier 10.1109/TMM.2020.3038311



(a) RGB image

(b) Ground truth

(c) Multi-branch [9]

(d) Ours

Fig. 1. An example of gaze prediction while driving. The results of (c) Multi-branch and (d) ours are produced with the input (a) a video clip. Comparing with (c) the result of Multi-branch, (d) the predicted map of ours is more concentrated and more accurate.

prediction maps, in which gaze tends to appear in tightly clustered areas.

In this paper, we introduce Reinforced Attention (RA) to get the attention mentioned above. The attention localization can be solved by a regression method (*i.e.*, Recurrent Neural Network (RNN)). Given a sequence of frames, we can directly apply a standard RNN to produce an attention location. In RNN, we need to apply convolution for the entire frames. However, this process is computationally expensive, as the computational cost scales linearly with the number of image pixels. While down-sampling the input frames can reduce the computational burden, many local details (*e.g. lanes and signs*) are lost in the process. Instead, we propose to use a reinforced attention model to estimate the attention, which we term reinforced attention. Our method has the following characteristics: 1) The backbone of RA is a recurrent neural network where the input frames are processed sequentially. 2) RA selectively chooses parts of the images to process to save computational resources. At each frame, the model has a sampler to select the next patch to sample based on the previous internal state. At the final frame, the model makes the decision on where is the attention. The above procedure uses Williams's REINFORCE [13] to address the non-differentiates due to the control problem.

There are two main advantages of RA. On the one hand, RA solves the gaze prediction problem in a way of reinforcement learning. This method is derived from the real driving scene, where a driver makes observations and chooses actions. This process is very similar to the process of reinforcement learning, which makes a sequence of decisions in a dynamic environment. On the other hand, RA achieves competitive results at a lower computational cost. Previous methods take all video content as input and apply convolution for the entire frames, making them prohibitively slow and expensive. However, RA selectively chooses parts of the images to process and spends fewer computing resources.

In addition, we also introduce speed and course details into the reinforced model. Previous works [9] only analyse the relationship between the gaze locations and speeds, and do not make use of speed and course details to train their gaze prediction model. In contrast, our method uses these details to improve the gaze prediction accuracy. Specifically, we feed these details as well as the patches obtained by the reinforced model into the RA model.

Overall, we propose a driver gaze prediction method by introducing reinforced attention. The overall framework can be described as follows. First, we use Multi-branch [9] to obtain an initial prediction. Multi-branch integrates three sources of information: raw video, motion and scene semantics. We use the predicted gaze map as our baseline. Then, RA is employed to estimate the attention location for the input video. Finally, we combine the gaze prediction map and the attention to generate the final result.

In summary, this paper makes the following main points.

- Based on existing gaze prediction approaches, we introduce RA into the framework to estimate the attention. Compared with previous methods, the proposed approach can produce more clustered and accurate predictions.
- RA is a reinforcement learning strategy, which servers as a location prediction policy and achieves competitive results at a lower computational cost.
- We argue that speed and course details are beneficial cues for attention localization. To the best of our knowledge, we are the first to introduce speed and course into the attention localization task.
- Extensive experiments confirm the consistent effectiveness of RA and the overall framework both quantitatively and qualitatively. Meanwhile, we provide ablation studies to show the contributions of different components.

II. Related Work

Appearance-based gaze prediction: There are two kinds of gaze prediction tasks, appearance-based gaze prediction and scene-based gaze prediction. Appearance-based gaze prediction takes images of human face and annotated eye gaze as input to learn a direct image-to-gaze mapping. Feng *et al.* [14] propose a hidden Markov model based gaze prediction system that utilizes the visual saliency of the content being viewed. Davies *et al.* [15] present a multicue gaze prediction framework for open signed video content, and inverstigate which cues are relevant for gaze prediction. Kellnhofer *et al.* [16] present Gaze360, a large-scale gaze-tracking dataset. In [16], the authors propose a 3D gaze model that includes temporal information and output the probability of gaze prediction. Other deep learning methods employed to solve the appearance-based gaze prediction task are described in [17]–[19]. These methods need to apply eye or face detection before training. The appearance-based task is primarily studied as a behavioral cue to better understand human thought processes.

Scene-based gaze prediction: In scene-based gaze prediction, images or videos of the outside environment are taken as input. One way to address this task is to use saliency prediction approaches [20]–[23]. Zhang *et al.* [22] propose a SP-MIL framework for co-saliency detection, which integrates both multiple instance learning and self-paced learning into a unified learning framework. Han *et al.* [23] propose a bottom-up salient object detection framework based on the background prior. In [23], the proposed method adopts a stacked denoising autoencoder to learn powerful representations for saliency computation. Since Alletto *et al.* [5] propose the DR(eye)VE dataset, the works related to driver gaze prediction begin to appear. Footage for the DR(eye)VE dataset is recorded as individuals are driving and the driver gaze is also saved. Palazzi *et al.* [4] model the driver gaze by training a coarse-to-fine convolutional network on short video clips from the DR(eye)VE dataset. In [9], the authors propose a complementary model based on a deep multi-branch architecture. This model integrates three sources of information: raw video, motion cues (in terms of optical flow) and scene semantics. This work focuses on the task of scene-based gaze prediction. The input is 16 raw video frames and the output is a gaze prediction map. Compared with appearance-based methods, the scene-based method primarily studies the relationship between the drivers and the scenes they are viewing. In this work, we focus on the scene-based gaze prediction.

Assisted driving: Gaze prediction tasks are primarily studied in the context of assisted driving [24], [25]. One challenge in assisted driving is the comprehension of the current traffic situation and the anticipation of all traffic participants' future driving behavior. To address this problem, Rehder *et al.* [26] propose a prediction framework that is able to infer a driver's maneuver intention. In [27], the authors model traffic phenomena by taking into account safety-relevant processes and perform "stochastic" simulation on large, representative virtual samples. In this way, safety performance of assisted and automated driving can be quantified by virtual experiments. Simon *et al.* [28] present an improvement of the advanced driver assistance systems by estimating the saliency of road signs using SVM learning techniques. Bredmond *et al.* [29] review a cluster of visual attention studies and conduct more realistic experiments with a larger set of targets, including pedestrians and bicycles. Pugeault *et al.* [30] propose a novel vision-based method that predicts driver behavior in real-time. They find that the field of view used by the computational model is closely related to driver gaze locations. In addition, Li *et al.* [24] focus on monitoring the driver attention level and propose a driver monitoring system that is able to sense inattentive drivers.

Deep reinforcement learning: Our method utilizes deep reinforcement learning (DRL) to locate the regions of attention. DRL is a framework in which decision-making networks interact with an environment and seek to learn a policy to take actions that maximise an environment reward. Similar to deep learning methods [31]–[33], DRL has been applied to many computer vision tasks [34]–[37], including bounding box location prediction [38], image caption [39], and seeding points for segmentation [40]. In [36], Takanobu *et al.* propose a reinforcement learning method for topic segmentation and labeling in goal-oriented dialogues. They attempt to solve the task in a weakly supervised setting and formulate it as a sequential decision problem. Uzkent and Ermon [37] propose a reinforcement learning approach to selectively use high resolution data when necessary while maintaining accuracy and reducing acquisition/run-time cost. Several works utilize DRL to predict the focus of human attention. Minut *et al.* [41] propose a model of selective attention for visual search tasks and introduce a reinforcement learning framework for sequential decision-making. Mnih *et al.* [42] present a novel recurrent neural network model for classification task. This method can extract information from an image by adaptively selecting a sequence of regions or locations and processing the selected regions only with high resolution. In [43], the authors propose a Dueling Network, which represents two separate estimators: one for the state value function and one for the state-dependent action advantage function. They also illustrate that the area of the network paying attention to is reasonable and important. Xu *et al.* [44] apply A3C [45] to predict head movement positions and focus of attention. Inspired by these reinforced methods, we train a network based on Williams's REINFORCE [13] to learn the policy in continuous action space to produce the driver attention.

III. The Proposed Method

In this section, we review the Multi-branch [9] model, which is applied to predict the driver gaze in Section III-A. We then describe the process of acquiring the reinforced attention in Section III-B. After getting the initial gaze prediction map and the attention, we employ mask-guided multiplication, in Section III-C, to generate the final prediction.

A. Multi-Branch Review

In this work, we employ the Multi-branch [9] model to get an initial gaze prediction map. The input of Multi-branch is a video clip with 16 frames. The Multi-branch model is composed of three different branches: RGB image (I branch), optical flow (F branch) and semantic segmentation (S branch). Each branch exploits complementary details which contribute to the final prediction.

The three branches have the same architecture. Each branch is a two-input two-output architecture composed of two streams, the cropped stream and the resized stream. The two inputs are fed into a weight shared C3D [46] model which has been pre-trained. The resized stream differs from the cropped stream due to a set of refine layers following the C3D model. The prediction of the resized input is stacked with the last frame of the video clip and then fed to the refine layers. The refine layers then process and upsample the tensor back to the input spatial resolution.

The model is trained in two steps. The first is single branch training in which each branch is trained separately using the aforementioned method. Following this, the three branches are then simultaneously fine-tuned. Prediction cost is minimized in terms of Kullback-Leibler divergence:

$$D_{KL}(Y||\hat{Y}) = \sum_i Y(i) \log\left(\epsilon + \frac{Y(i)}{\epsilon + \hat{Y}(i)}\right), \tag{1}$$

where $\hat{Y}$ is the prediction map of Multi-branch and Y is the ground truth. i is the summation index that spans across image pixels and ϵ denotes a small constant.

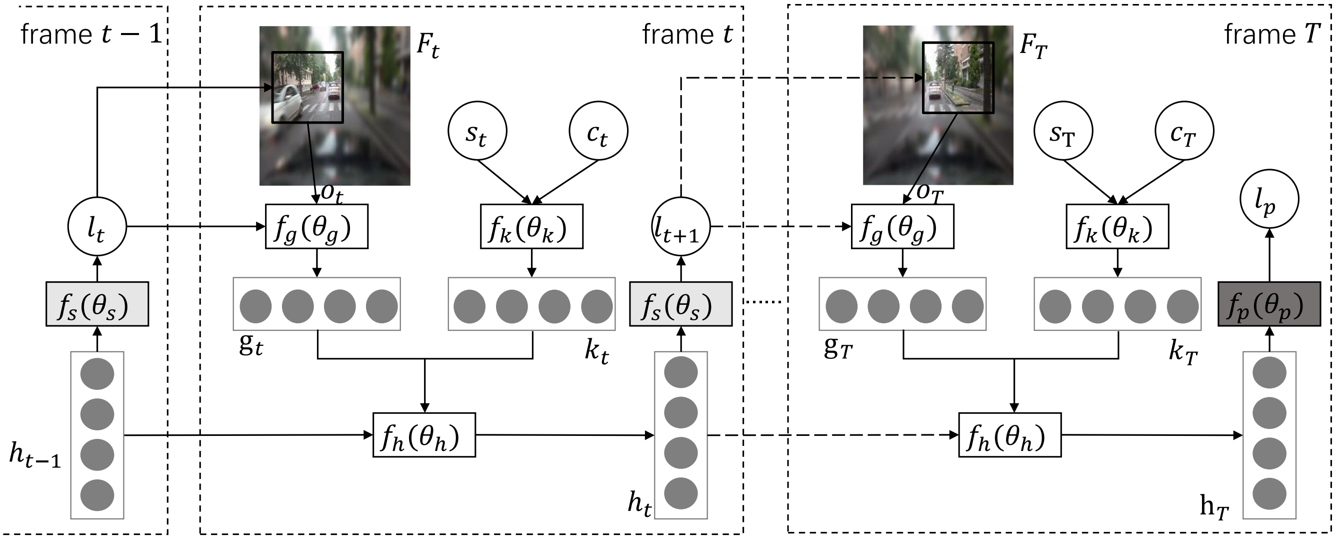


Fig. 2. Architecture of the *reinforced attention* (RA) model. Our model is based on Recurrent Neural Network (RNN) and the RNN iteration is repeated for T steps. *1) At Frame* t, the core network $f_h(\theta_h)$ takes multiple features as input: observation feature g_t, speed-course feature k_t and the internal feature h_{t-1}. Note that h_{t-1} is produced at the previous time step $t-1$. *2) At the last Frame* T, the model produces the final location l_p. Instead of applying $f_s(\theta_s)$ to generate a sample location, $f_p(\theta_p)$ is applied to produce the final attention location l_p of the video clip. Note that $f_p(\theta_p)$ and $f_s(\theta_s)$ have the same architecture but different parameters.

B. Reinforced Attention

This section presents how the Reinforced Attention (RA) obtains the attention. Fig. 2 shows the overall framework of RA, in which *REINFORCE* [47] is embedded to generate the reinforced attention.

In a typical DRL-based method, there exists an agent taking actions in an environment to maximize the cumulative rewards. The process by which an agent makes decisions is called policy π. π can be a deep network and map states to the action with some probability. The environment can be modelled as a Markov decision process, where the current state and action depend only on previous state and action. Given an observation o_t at each time step t, the agent chooses an action a_t and receives a reward R_t. The reward R_t at each step t is either a future or discounted reward, which can be described as $R_t = \sum_{i=t}^{T} \gamma^{i-t} r(s_i, a_i)$. $r(s_i, a_i)$ indicates the reward at state s_i when taking action a_i. $\gamma \in [0, 1]$ is the discount-rate. As a particular state becomes older, its effect on the later states becomes less and less. Thus, this reward is discounted by γ, which is less than 1.

We formulate the driver gaze prediction task in an RL framework as shown in Fig. 2. The input of the attention model is video frames $\{F_t\}_{t=1}^{T}$ with frame number t ranging from 1 to T and the output is a predicted location l_p, where the attention is located.

Actions: The parameters of the RA agent are composed of several networks described as: $\{\theta_g, \theta_k, \theta_h, \theta_s, \theta_p\}$. Here $f_h(\theta_h)$ is the core network. As shown in Fig. 2, the core network at time t takes three kinds of inputs: the localization and observation feature g_t, the speed-course feature k_t and the internal representation h_{t-1} at previous time step. $f_s(\theta_s)$ has the same architecture with $f_p(\theta_p)$, which is applied to produce a location. The difference is that $f_s(\theta_s)$ is a sampler deciding where to sample the patch at frame $t+1$, while $f_p(\theta_p)$ determines the final action. This final action is the predicted location of attention for the input video clip. Our objective is to maximize the expectation of accumulated future rewards:

$$J(\theta) = \mathbb{E}_{p(s_{1:T};\theta)}\left[\sum_{t=1}^{T} r_t\right] = E_{p(s_{1:T};\theta)}[R], \tag{2}$$

where $s_{1:T}$ is a possible interaction sequences and $p(s_{1:T};\theta)$ is the probability of $s_{1:T}$. Here $p(s_1 : T; \theta)$ is based on current policy θ. During training, We use REINFORCE [47] to calculate the gradient,

$$\nabla_\theta J = \sum_{t=1}^{T} \mathbb{E}_{p(s_{1:T};\theta)}[\nabla_\theta \log \pi(a_t|s_{1:t};\theta)R]$$
$$\approx \frac{1}{M}\sum_{i=1}^{M}\sum_{t=1}^{T} \nabla_\theta \log \pi(a_t^i|s_{1:t}^i;\theta)R^i, \tag{3}$$

where a_t is the action we take in time step t.

Rewards: One of the major tasks in training the RA network is formulating the reward function. A reward R reflects the quality of the action and is given back to the agent. In the case of RA, R is determined by the ground truth attention $l_g = (x_g, y_g)$ and final action $l_p = (x_p, y_p)$, which is produced by $f_p(\theta_p)$. We use Euclidean distance $d(\cdot, \cdot)$ to calculate the distance between two points and the reward function R is defined as:

$$R = 1 - d(l_p, l_g). \tag{4}$$

Training the RA model. In training the RA model, the actor interacts with the environment. The interaction is achieved in our RA model through the following procedure:

1) At frame $t \in [1, ..., T-1]$, the sampler obtains the current observation o_t from the input frame F_t, according to

the position l_t. Note that l_t is generated by processing the previous frames. More specifically, o_t is a square with a side length of 64 pixels centered at l_t.

2) Multiple details from the current frame t and the RNN feature h_{t-1} in the last frame are fed into the core network $f_h(\theta_h)$. These details of current frame include the observation o_t, the location l_t, speed s_t and course c_t. As shown in Fig. 2, the RA model contains several deep modules, which are to extract the features described above. The speed and course details are represented by two normalized numbers. The normalized values are fed into $f_k(\theta_k)$ to get the speed and course feature k_t. Note that $f_k(\theta_k)$ is a fully connected layer.

3) The RA model produces the RNN feature h_t by $f_h(\theta_h)$. Based on h_t, the sample network $f_s(\theta_s)$ produces the next sample location l_{t+1}. Note that h_t and l_{t+1} will be delivered to next frame $t+1$.

4) When arriving at the final frame $t = T$, RA uses action $f_p(\theta_p)$ to output the final location of attention. $f_p(\theta_p)$ and $f_s(\theta_s)$ have the same network architecture but different parameters. This is primarily because $f_s(\theta_s)$ is a sampler which mines useful details, including the sample location and the corresponding observation. However, $f_s(\theta_s)$ is employed to make final decisions based on the driving state collected from the video clip.

5) Once the RA model meets the termination condition of giving the final decision l_p, all experiences, together with the rewards R are delivered to the optimizer to upgrade the RA network.

C. Prediction with Reinforced Attention

In this section, we propose mask-guided multiplication to generate the final gaze prediction. After obtaining the reinforced attention, we need to combine it with the fixation map F described in Section III-A. The fixation map is a $W \times H$ distribution matrix and the reinforced attention l_p is a location. Thus, we generate a mask M by using the attention location l_p. Here M is 2-D gaussian distribution with mean (x_p, y_p) and $\sigma = 0.1$. Following this, each fixation map, F, generated by *Multi-branch* is masked with the learned attention mask M. The final gaze prediction map $\hat{F}$ is defined as:

$$\hat{F} = F \odot M, \tag{5}$$

where $\odot$ denotes element-wise multiplication.

D. Discussion

The difference between the RA model and the typical RNN model. Attention localization can be regarded as a regression task with video as input and a prediction map as output. Typically this is achieved by feeding all video frames into the RNN model and then processing them with convolution filter maps. This process is computationally expensive, because all the pixels would be processed by convolutional process.

In contrast to the typical RNN, the proposed RA model adaptively selects a region to process at high resolution. Comparing with RNN, RA process less pixels and thus is more computational efficient. This technique is inspired by the perception mechanisms of human drivers who do not process a scene in full. Experiment in Section IV-E indicates that RA has a competitive accuracy when compared with the typical RNN. The main reason is that continuous frames contain much redundant information, it is unnecessary to process the entire frame at each step.

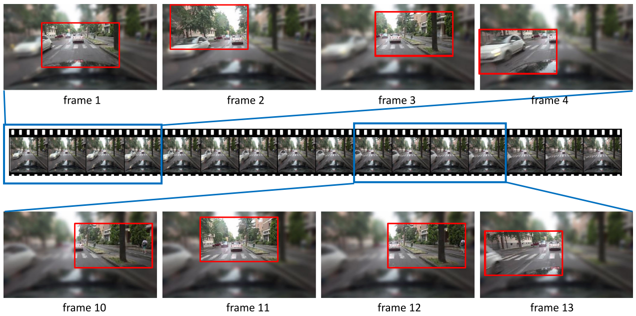


Fig. 3. A demonstration of the selective process mechanism. As the video plays sequentially, the sampler $f_s(\theta_s)$ chooses a patch rather than processing the entire frame. At each time step, the sampler has observed several patches in previous steps. Based on these patches, the sampler determines where to locate to get more details. After the sampler obtains the patch to be processed, the sampler moves on to the next frame.

The mechanism of the sampler: We illustrate the sampling process in Fig. 3. Taking $T = 16$ frames as input, the sampler selects a specific area for each frame to process. As the frames move forward, the sampler is found to select different locations to process. A possible explanation is that the small differences between successive frames lead the sampler to explore other areas in which it can collect new details. This can also be explained by the fact that the reinforced model is fed with appearance and location features so the sampler can perceive which areas have been collected and which are not.

IV. Experiment

A. Datasets and Evaluation Metrics

Datasets: We evaluate our method on DR(eye)VE [5], which is the first publicly available dataset addressing driver gaze prediction. While there exist several dataset for gaze prediction [16], [18], [48], only DR(eye)VE is setup for scene-based driver gaze prediction, the rest being designed for appearance-based gaze prediction tasks.

The DR(eye)VE dataset consists of 74 sequences with 555,000 frames. Each sequence is 5 minutes long and is captured at 1080p/25fps. Additionally, the dataset includes other relevant driving state such as GPS data, accelerometer and gyroscope measurements. In this work, we mainly concern the speed and course details, which are then employed in our reinforced attention model.

Moreover, we evaluate the methods on the complete test set as well as the acting subset. The acting subset is particularly interesting as the deviation of driver gaze from central pattern denotes an intention related to some driving actions (*e.g.* changing lanes and overtaking).

Evaluation metrics: To evaluate the proposed method, we compare our approach with the state-of-the-art methods primarily in two aspects, *i.e.,* the accuracy of gaze prediction and the

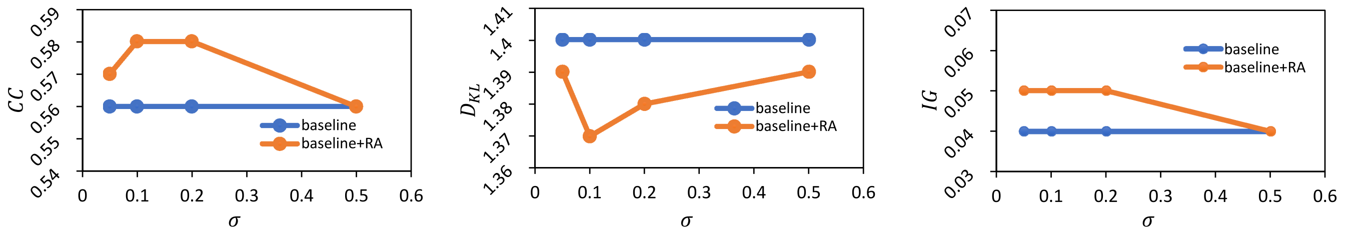


Fig. 4. Evaluation with different σ on the complete dataset of DR(eye)VE. By adopting RA, baseline+RA achieves better results than baseline in terms of CC, D_{KL} and IG.

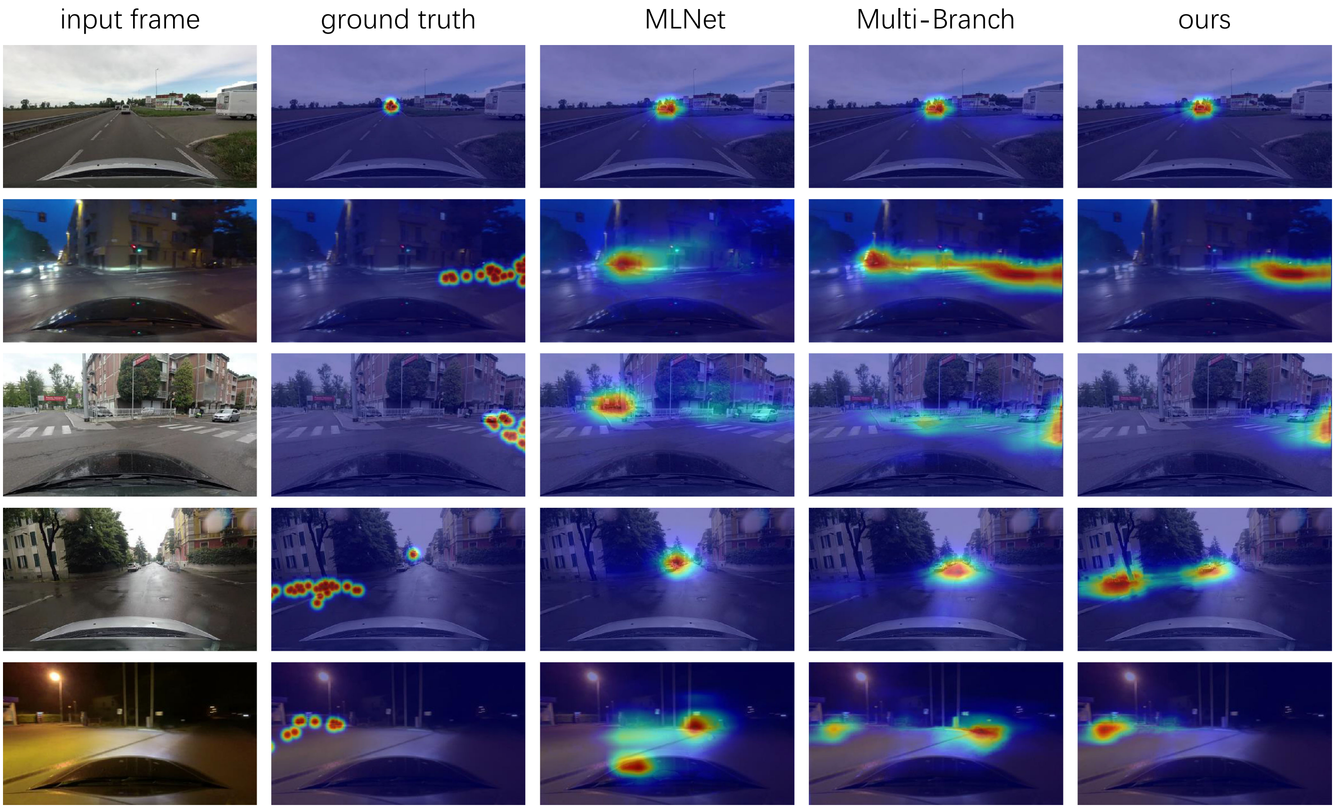


Fig. 5. Qualitative examples of the gaze prediction maps. Each row is a set of examples. Each test sample is represented by five images: input raw image, ground truth map, prediction of MLNet [20], prediction of Multi-branch [9] and our prediction. The results show that the prediction maps produced by our method are more focused (the first three rows) and more accurate (the last two rows).

accuracy of reinforced attention. We evaluate the gaze prediction result following the guidelines in [9], [49]. Specifically, we use Person's Correlation Coefficient (CC), Kullback-Leibler Divergence (D_{KL}) and Information Gain (IG) for evaluation on DR(eye)VE. For CC and IG, higher is better, while for D_{KL} the opposite is true. IG is a measure of the quality of a predicted map P with respect to a ground truth map Y in presence of a strong bias,

$$IG(P, Y, B) = \frac{1}{N} \sum_{i} Y_i[\log_2(\epsilon + P_i) - \log_2(\epsilon + B_i)], \quad (6)$$

where i is an index spanning all the N pixels in the image and ϵ is a pre-defined constant to ensure numerical stability. Note that B is the bias map computed by averaging the training fixation map. The ground truth map Y for a frame is built by accumulating the gaze points of the nearby 25 frames [9]. For evaluating the reinforced attention model, we use the Euclidean distance $d(l_p, l_g)$ between two locations to evaluate the distance between the predicted location l_p with ground truth attention l_g. l_g is a location, which is obtained by calculating the gaze prediction ground truth Y. As shown in Fig. 5, for each ground truth Y (column 2), there exist several gathered points. All the points are Gaussian distribution and they have the same maximum value. In this work, the ground truth attention l_g is defined by averaging the coordinates of these maximum points:

$$l_g = \frac{1}{N} \sum_{n=1}^{N} l_n, \quad (7)$$

where N is the number of maximum value points in Y and (x_n, y_n) is the coordinate of n-th point l_n. Note that $(x_n, y_n) \in [-1, 1]$.

TABLE I
COMPARISON WITH STATE-OF-THE-ART ON THE COMPLETE TEST SET OF DR(EYE)VE

Method	$CC \uparrow$	$D_{KL} \downarrow$	$IG \uparrow$
Baseline Gaussian	0.40	2.16	-0.49
Baseline Mean	0.51	1.60	0.00
Mathe *et al.* [10]	0.04	3.30	-2.08
Wang *et al.* [50]	0.04	3.40	-2.21
Wang *et al.* [51]	0.11	3.06	-1.72
MLNet [20]	0.44	2.00	-0.88
RMDN [21]	0.41	1.77	-0.06
SAM [52]	0.38	2.56	-1.23
Palazzi *et al.* [4]	0.55	1.48	-0.21
Multi-branch [9]	0.56	1.40	0.04
Ours	**0.58** ± 0.008	**1.37** ± 0.004	**0.05** ± 0.016

TABLE II
COMPARISON WITH STATE-OF-THE-ART ON THE ACTING SUBSEQUENCES OF DR(EYE)VE

Method	$CC \uparrow$	$D_{KL} \downarrow$	$IG \uparrow$
Baseline Gaussian	0.26	2.41	0.03
Baseline Mean	0.22	2.35	0.00
MLNet [20]	0.32	2.35	-0.36
RMDN [21]	0.31	2.13	0.31
Palazzi *et al.* [4]	0.37	2.00	0.20
Multi-branch [9]	0.41	1.80	0.51
Ours	**0.42** ± 0.016	**1.79** ± 0.004	**0.51** ± 0.008

B. Experiment Setting

Baseline model for gaze prediction. We use Multi-branch as the baseline of our overall framework. To train the baseline, we follow the training strategy in [9]. Specifically, the Multi-branch model is split into three branches where each branch is fed with 16 frames clips in raw RGB color space, 16 frames clips with optical flow maps and 16 frames clips with semantic segmentation maps, respectively. The training process is divided into two stages: training the three branches independently and then fine-tuning the complete Multi-branch model with a lower learning rate value. More details can be found in [9].

Reinforced attention model: We train the reinforced attention model following Sec. III-B. In our implementation, input frames are resized to 256×256 and the sampled patch has the size of 64. The dimension of the internal state vector h_t, appearance feature g_t and speed-course feature k_t is 256. The model is trained via stochastic gradient descent with batch size 32 and momentum of 0.9. The learning rate is initialised at 0.003.

C. Parameters Analysis

An important hyper-parameter of RA is σ described in Section III-C. This parameter is used to generate the weighted mask M. We show its impact by varying its value in Fig. 4. In this part, we can draw two conclusions. On the one hand, we observe that our method with different σ consistently improves the baseline. As a regulatory mechanism, RA provides accurate attention and help generate more clustered and accurate prediction maps. On the other hand, baseline+RA arrives the best performance when $\sigma = 0.1$ with 0.58 CC, 1.37 D_{KL} and 0.05 IG. In this work, a low σ indicates that the gaze prediction values tend to be close to the focus of attention, while a high σ indicates that the values are spread out over a wider range. When σ is set to 0.1, the gaze prediction maps are reasonably adjusted. Note that we set $\sigma = 0.1$ in the following experiments.

D. Comparison With State-of-the-Art Methods

We compare our approach with state-of-the-art methods on the complete test set and the acting subsequences of DR(eye)VE. Note that we use the same model to test on the two sets. Table I and II report the comparison when tested on the complete set and the acting set, respectively. We compare with two baseline methods: *Baseline Gaussian* and *Baseline Mean*, several saliency detection based methods: MLNet [20] and RMDN [21], and previous gaze prediction methods: Palazzi *et al.* [4] and Multi-branch [9].

We first compare with two baseline methods which do not require training. The *Baseline Gaussian* method is employed by a centered gaussian baseline and the *Baseline Mean* is generated by averaging all training set prediction maps. On the complete test set and the acting subsequences, the proposed method outperforms the *Baseline Gaussian* and the *Baseline Mean*. For example, the D_{KL} value of *Baseline Gaussian* and *Baseline Mean* are 2.16 and 1.60, respectively, both higher than the value of our method. This indicates that our method can deal with task-driven changes in gaze prediction, even if the gaze distribution is often strongly biased to the vanishing point of the road.

Next, we compare with several saliency prediction based methods, which can be employed to solve gaze prediction task. As shown in Fig. I, RMDN [21] yields a D_{KL} of 1.77 on the test set. It is lower than the Baseline Gaussian but is higher than Palazzi *et al.* [4] by 0.29. This is probably due to that [4] is a task-orientated method designed for gaze prediction, while RMDN is not. In addition, we compare the proposed method with the LSTM-based Saliency Attentive Model (SAM) [52], which aims at predicting where human gazes will focus on a given image. The D_{KL} value of SAM is 2.56, higher than the value of the proposed method. It shows that our method outperforms SAM. The main reason is that SAM takes an image as input and is not designed for video-based tasks.

We also evaluate the methods on the complete set and the acting subset. We observe that the results of *Baseline Gaussian* and the *Baseline Mean* performed on the complete set are not as accurate as those performed on the acting set. For example, the *Baseline Gaussian* achieves $D_{KL} = 2.16$ when tested on complete set, and obtains $D_{KL} = 2.41$ on when tested on the acting set. It indicates that the deviation of driver gaze from the central pattern does occur when undergoing task-specific actions (*e.g. changing lanes and overtaking*). One possible solution is to introduce attention mechanism, which can highlight the valuable ares. Results show that our method provide more accurate gaze maps on both sets.

Comparing with state-of-the-art gaze prediction methods, our approach clearly achieves higher performance on both sets. Specifically, our method achieves $D_{KL} = 1.37$ and $CC = 0.58$ on complete set, and obtains $D_{KL} = 1.79$ and $CC = 0.42$ on

TABLE III
COMPARISON OF ATTENTION LOCALIZATION METHODS ON THE COMPLETE SET OF DR(EYE)VE. NOTE THAT S. INDICATES SPEED AND C. INDICATES COURSE

Method	$L_2 \downarrow$	$CC \uparrow$	$D_{KL} \downarrow$	$IG \uparrow$
RNN *w/o* S. & C.	0.27	0.51	1.42	0.01
RNN *w/o* S.	0.22	0.56	1.38	0.03
RNN *w/o* C.	0.27	0.54	1.40	0.02
RNN	0.20	0.57	1.38	**0.06**
RA *w/o* S. & C.	0.26	0.52	1.41	0.01
RA *w/o* S.	0.21	0.57	1.37	0.05
RA *w/o* C.	0.24	0.55	1.38	0.04
RA	**0.20**	**0.58**	**1.37**	0.05

acting set. The D_{KL} value is 0.03 and 0.01 lower than the current best results (Multi-branch [9]) when tested on the complete set and the acting set, respectively. The above results show that our method can quantitatively improve the gaze prediction accuracy. The main reason is that the proposed method is a good complementary to the existing methods. RA is built on top of existing methods, providing an accurate attention for gaze prediction maps. This attention can regulate the maps and output more clustered and accurate results. Qualitatively results are shown in Fig. 5. The results illustrate that our method can generate more concentrated and more accurate gaze maps.

E. Evaluation of Reinforced Attention

Compared with RNN, our approach is competitive in accuracy: In this section, we evaluate the reinforced attention (RA) model. The results are shown in Table III. On the DR(eye)VE dataset, we compare RA with typical recurrent neural network (RNN). The RNN takes a set of images as input and can be viewed as a standard regression problem. On the complete test set, the proposed RA yields an L_2 of 0.20, while the L_2 of RNN is also 0.20. Comparing with RNN, in some indicators, our method outperforms RNN. Specifically, our results are higher by 0.01 in CC and lower by -0.01 in D_{KL}. However, RNN yields an IG of 0.06, higher than RA. The results indicate that RA yields competitive results on the complete set of DR(eye)VE.

Compared with RNN, our approach is more computational efficient: To quantitatively measure the computational cost of RA, we use the number of model parameters and the inference time for evaluation. In this paper, RNN model has 32.4 million parameters, while RA only has 24.8 million parameters. For inference time, we test the two models on a GPU of NVIDIA 2080 Ti. In testing phase, RNN takes 1.2s per mini-batch, while RA only takes 0.5s. Note that, the batch size is set to 32. The results indicate that our approach is more computational efficient. To process a video clip, RNN needs to apply convolution operation for the entire frames, while RA selectively chooses a patch of each frame. The patch is one quarter of the size of the entire image. It indicates that the computational cost of RA is much lower than that of RNN.

Speed and course details are useful for attention localization: This section also presents ablation studies of the RA model. Since the driving condition details are involved, *i.e.*, speed and course, we remove them one at a time to evaluate their contribution respectively. Results on DR(eye)VE are shown in Table III. When removing speed and course from the RA system, L_2 and D_{KL} will be 0.06 and 0.04 higher than the full RA model in the complete set. Meanwhile, the L_2 of RNN is 0.20, which is also lower than RNN *w/o* speed and course. Recall that lower L_2 is better. These results show that speed and course details play an important role in estimating the driver attention.

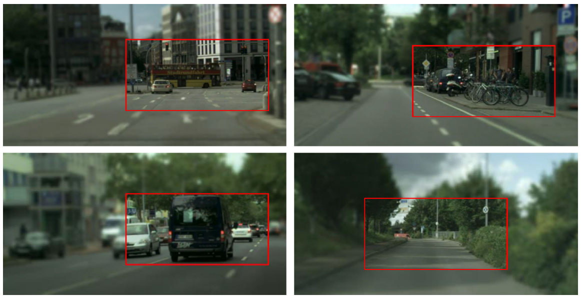


Fig. 6. Examples of the reinfoced attention on the CityScapes dataset [53]. The patches in red rectangle are the attentions predicted by our method.

Course details contribute more than speed details: To further evaluate the respective importance of speed and course details, we add another experiment. we remove speed or course details, one at a time from the system. Overall, the models *w/o* speed or *w/o* course achieve lower L_2 and D_{KL} than the ones *w/o* speed and course, but achieve higher L_2 and D_{KL} than the complete system. For example, RA *w/o* course yields 0.24 in L_2 and 1.38 in D_{KL}, both slightly lower than RA *w/o* speed and course, but higher than complete RA. We also find that the course detail is more import than the speed. RA *w/o* speed yields 0.21 in L_2 and 1.37 in D_{KL}, which are lower than RA *w/o* course.

RA performs well on CityScapes [53]: To evaluate the effectiveness of RA, we also test our method on the CityScapes dataset. The results are shown in Fig. 6. Similar to DR(eye)VE, the CityScapes dataset is comprised of a large set of video sequences recorded in streets. Note that the CityScapes dataset is designed to address the task of semantic segmentation. However, it is valuable to utilize the videos of CityScapes to validate the effectiveness of RA. The results indicate that RA can output reasonable attentions. In Fig 6, the examples show that RA focus on the objects that can affect driving, *e.g.,* moving vehicles, roadside bicycles, and vanishing points of the road.

F. Variant Study.

We further evaluate three different variants of Multi-branch. *i.e.*, Image Branch, Flow Branch and Segmentation Branch. As mentioned above, our method is built directly on top of existing methods. Thus, we use other methods as our baseline. As described in Section III-A, image branch, flow branch and segmentation branch are parts of the Multi-branch model. Table IV details the CC, D_{KL} and IG results of each variant in the DR(eye)VE complete set. *I* refers to the RGB image branch. As the result shows, *I+RA* yields 1.40 in D_{KL}, which is slightly lower than the baseline *I* 1.41. However, compared with *S* (1.69), *S+RA* (1.59) significantly improves the result by 0.1 in terms of D_{KL}. Meanwhile, when using the RA model, the D_{KL} value of *(I+F+S)+RA* will drop 0.03, compared with *I+F+S*. These results prove that the RA model has the ability to improve the gaze prediction accuracy on top of existing methods.

TABLE IV
VARIANT STUDY OF OUR METHOD ON THE COMPLETE TEST SET OF DR(EYE)VE. I, F, AND S REPRESENT IMAGE, OPTICAL FLOW AND SEMANTIC SEGMENTATION BRANCHES, RESPECTIVELY. (IFS) EQUALS THE MULTI-BRANCH MODEL. RA MEANS THE REINFORCED ATTENTION

Method	*CC* ↑	D_{KL} ↓	*IG* ↑
I	0.55	1.41	-0.01
F	0.51	1.61	-0.13
S	0.47	1.69	-0.11
IFS	0.56	1.40	0.04
I+RA	0.56(+0.01)	1.40(-0.01)	0.01(+0.02)
F+RA	0.52(+0.01)	1.55(-0.06)	-0.12(+0.01)
S+RA	0.50(+0.03)	1.59(-0.10)	-0.10(+0.01)
IFS+RA	**0.58 (+0.02)**	**1.37(-0.03)**	**0.05(+0.01)**

V. CONCLUSION

In this paper, we propose to use attention to improve the driver gaze prediction task. Based on existing gaze prediction approaches, we use attention to regulate the initial predicted results to obtain more concentrated and accurate gaze maps. Specifically, we propose a reinforcement learning based model, termed Reinforced Attention (RA), for attention localization. RA is able to produce competitive localization accuracy while only processing a small subset of the video. When training RA, we also feed RA with speed and course details, which are proven to be indispensable. Experiments on the complete test set and the acting set of DR(eye)VE show that our method yields consistent improvement over several baselines and compares favorably with the state-of-the-art approaches.

ACKNOWLEDGMENT

The authors would like to thank for the support from HAWK-EYE Group.

REFERENCES

[1] S. Habenicht, H. Winner, S. Bone, F. Sasse, and P. Korzenietz, “A maneuver-based lane change assistance system,” in *Proc. IEEE Intell. Vehicles Symp.*, 2011, pp. 375–380.

[2] L. Chen, Q. Li, M. Li, and Q. Mao, “Traffic sign detection and recognition for intelligent vehicle,” in *Proc. IEEE Intell. Vehicles Symp.*, 2011, pp. 908–913.

[3] K.-H. Lee and J.-N. Hwang, “On-road pedestrian tracking across multiple driving recorders,” *IEEE Trans. Multimedia*, vol. 17, no. 9, pp. 1429–1438, Sep. 2015.

[4] A. Palazzi, F. Solera, S. Calderara, S. Alletto, and R. Cucchiara, “Where should you attend while driving?” 2016, *arXiv:1611.08215*.

[5] S. Alletto, A. Palazzi, F. Solera, S. Calderara, and R. Cucchiara, “Dr (eye) ve: A dataset for attention-based tasks with applications to autonomous and assisted driving,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2016, pp. 54–60.

[6] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, “Frequency-tuned salient region detection,” in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, no. CONF, 2009, pp. 1597–1604.

[7] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, “Global contrast based salient region detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015.

[8] Y.-F. Ma and H.-J. Zhang, “Contrast-based image attention analysis by using fuzzy growing,” in *Proc. ACM Int. Conf. Multimedia*, 2003, pp. 374–381.

[9] A. Palazzi *et al.*, “Predicting the driver’s focus of attention: The DR (eye) VE project,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1720–1733, Jul. 2019.

[10] S. Mathe and C. Sminchisescu, “Actions in the eye: Dynamic gaze datasets and learnt saliency models for visual recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 7, pp. 1408–1424, Jul. 2015.

[11] S.-H. Zhong, Y. Liu, F. Ren, J. Zhang, and T. Ren, “Video saliency detection via dynamic consistent spatio-temporal attention modelling,” in *Proc. Twenty-seventh AAAI Conf. Artif. Intell.*, 2013, pp. 1063–1069.

[12] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning spatiotemporal features with 3D convolutional networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4489–4497.

[13] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, “Policy gradient methods for reinforcement learning with function approximation,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 1057–1063.

[14] Y. Feng, G. Cheung, W.-t. Tan, P. Le Callet, and Y. Ji, “Low-cost eye gaze prediction system for interactive networked video streaming,” *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1865–1879, Dec. 2013.

[15] S. J. Davies, D. Agrafiotis, C. N. Canagarajah, and D. R. Bull, “A multicue bayesian state estimator for gaze prediction in open signed video,” *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 39–48, Jan. 2009.

[16] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, “Gaze360: Physically unconstrained gaze estimation in the wild,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6912–6921.

[17] T. Fischer, H. Jin Chang, and Y. Demiris, “RT-GENE: Real-time eye gaze estimation in natural environments,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 334–352.

[18] K. Krafka *et al.*, “Eye tracking for everyone,” in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2176–2184.

[19] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, “Appearance-based gaze estimation in the wild,” in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4511–4520.

[20] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, “A deep multi-level network for saliency prediction,” in *Proc. Int. Conf. Pattern Recognit.*, 2016, pp. 3488–3493.

[21] L. Bazzani, H. Larochelle, and L. Torresani, “Recurrent mixture density network for spatiotemporal visual attention,” in *Proc. Int. Conf. Learn. Representations*, 2017.

[22] D. Zhang, D. Meng, and J. Han, “Co-saliency detection via a self-paced multiple-instance learning framework,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2017. [Online]. Available: https://doi.org/10.1109/TPAMI.2016.2567393

[23] J. Han *et al.*, “Background prior-based salient object detection via deep reconstruction residual,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 8, pp. 1309–1321, Aug. 2015. [Online]. Available: https://doi.org/10.1109/TCSVT.2014.2381471

[24] N. Li, J. J. Jain, and C. Busso, “Modeling of driver behavior in real world scenarios using multiple noninvasive sensors,” *IEEE Trans. Multimedia*, vol. 15, no. 5, pp. 1213–1225, Aug. 2013.

[25] J. Fang, D. Yan, J. Qiao, and J. Xue, “DADA: A large-scale benchmark and model for driver attention prediction in accidental scenarios,” *CoRR*, vol. abs/1912.12148, 2019. [Online]. Available: http://arxiv.org/abs/1912.12148

[26] T. Rehder, A. Koenig, M. Goehl, L. Louis, and D. Schramm, “Lane change intention awareness for assisted and automated driving on highways,” *IEEE Trans. Intell. Vehicles*, vol. 4, no. 2, pp. 265–276, Jun. 2019.

[27] T. Helmer, L. Wang, K. Kompass, and R. Kates, “Safety performance assessment of assisted and automated driving by virtual experiments: Stochastic microscopic traffic simulation as knowledge synthesis,” in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, 2015, pp. 2019–2023.

[28] L. Simon, J.-P. Tarel, and R. Brémond, “Alerting the drivers about road signs with poor visual saliency,” in *Proc. IEEE Intell. Vehicles Symp.*, 2009, pp. 48–53.

[29] R. Bremond *et al.*, “Where we look when we drive: A multidisciplinary approach,” in *Proc. Trans. Res. Arena*, Paris, France, Apr. 2014, Art. no. 10.

[30] N. Pugeault and R. Bowden, “How much of driving is preattentive?” *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5424–5438, Dec. 2015.

[31] K. Lv, H. Sheng, Z. Xiong, W. Li, and L. Zheng, “Pose-based view synthesis for vehicles: A perspective aware method,” *IEEE Trans. Image Process.*, vol. 29, pp. 5163–5174, 2020.

[32] H. Sheng *et al.*, “Mining hard samples globally and efficiently for person re-identification,” *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9611–9622, Oct. 2020.

[33] T. Zhang *et al.*, “A deep neural network-driven feature learning method for multi-view facial expression recognition,” *IEEE Trans. Multimedia*, vol. 18, no. 12, pp. 2528–2536, Dec. 2016.

[34] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, p. 529, 2015.

[35] Y. Zhang, X. Qian, X. Tan, J. Han, and Y. Tang, "Sketch-based image retrieval by salient contour reinforcement," *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1604–1615, Aug. 2016.

[36] R. Takanobu *et al.*, "A weakly supervised method for topic segmentation and labeling in goal-oriented dialogues via reinforcement learning," in *Proc. Twenty-Seventh Int. Joint Conf. Artif. Intell., IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, 7 2018, pp. 4403–4410. [Online]. Available: https://doi.org/10.24963/ijcai.2018/612

[37] B. Uzkent and S. Ermon, "Learning when and where to zoom with deep reinforcement learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 12342–12351.

[38] J. C. Caicedo and S. Lazebnik, "Active object localization with deep reinforcement learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2488–2496.

[39] N. Xu *et al.*, "Multi-level policy and reward-based deep reinforcement learning framework for image captioning," *IEEE Trans. Multimedia*, vol. 22, no. 5, pp. 1372–1383, May 2020.

[40] G. Song, H. Myeong, and K. Mu Lee, "SeedNet: Automatic seed generation with deep reinforcement learning for robust interactive segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1760–1768.

[41] S. Minut and S. Mahadevan, "A reinforcement learning model of selective visual attention," in *Proc. ACM Int. Conf. Auton. Agents*, 2001, pp. 457–464.

[42] V. Mnih *et al.*, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2204–2212.

[43] Z. Wang *et al.*, "Dueling network architectures for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1995–2003.

[44] M. Xu *et al.*, "Predicting head movement in panoramic video: A deep reinforcement learning approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2693–2708, Nov. 2019.

[45] V. Mnih *et al.*, "Asynchronous methods for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1928–1937.

[46] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4489–4497.

[47] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, no. 3-4, pp. 229–256, 1992.

[48] M. X. Huang, J. Li, G. Ngai, and H. V. Leong, "ScreenGlint: Practical, in-situ gaze estimation on smartphones," in *Proc. ACM Conf. Human Factors Comput. Syst.*, 2017, pp. 2546–2557.

[49] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, "What do different evaluation metrics tell us about saliency models?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 3, pp. 740–757, Mar. 2019.

[50] W. Wang, J. Shen, and F. Porikli, "Saliency-aware geodesic video object segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3395–3402.

[51] W. Wang, J. Shen, and L. Shao, "Consistent video saliency using local gradient flow optimization and global refinement," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4185–4196, Nov. 2015.

[52] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, "Predicting human eye fixations via an LSTM-based saliency attentive model," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5142–5154, Oct. 2018. [Online]. Available: https://doi.org/10.1109/TIP.2018.2851672

[53] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

Kai Lv received the B.S. degree from the School of Computer Science and Technology, Tianjin University of Science and Technology, Tianjin, China, in 2012. He is currently working toward the Ph.D. degree with the School of Computer Science and Engineering, Beihang University, Beijing, China.

Hao Sheng (Member, IEEE) received the B.S. and Ph.D. degrees from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2003 and 2009, respectively. He is currently a Professor and the Ph.D. Supervisor with the School of Computer Science and Engineering, Beihang University, China. He is currently working on computer vision, pattern recognition and machine learning.

Zhang Xiong received the B.S. degree from Harbin Engineering University, Harbin, China, in 1982, the M.S. degree from Beihang University, Beijing, China, in 1985. He is currently a Professor and the Ph.D. Supervisor with the School of Computer Science and Engineering, Beihang University, China. He is currently working on computer vision, information security and data vitalization.

Wei Li (Member, IEEE) graduated from the Department of Mathematics and Mechanics, Peking University, Beijing, China, in 1966 and the Ph.D. degree in computer science from the University of Edinburgh, Edinburgh, U.K., in 1983 He is currently a Computer Scientist. Since 1966 he has been a teacher with Beihang University (formerly Beijing Institute of Aeronautics), after four years of graduate study from the University of Edinburgh. He has been a Professor with the School of Computer Science and Engineering, Beihang University, since 1986 and served as a President with Beihang University from 2002 to 2009. He was elected as a member of the Chinese Academy of Sciences in 1997. He Currently serves as the Director of the State Key Laboratory of Software Development Environment, member of the National Educational Advisory Committee.

Liang Zheng (Member, IEEE) received B.E. degree in life science in 2010 and the Ph.D. degree in electronic engineering, in 2015 both from Tsinghua University, Beijing, China. He is currently a Lecturer and a Computer Science Futures Fellow with the Research School of Computer Science, Australian National University. He was a Postdoc Researcher with the Centre for Artificial Intelligence, University of Technology Sydney, Sydney, Australia. His research interests include image retrieval, classification, and person re-identification.