CGI2018 manuscript No.
(will be inserted by the editor)

Facial Landmarks based Region-Level Data Augmentation for Gaze Estimation

Abstract Data augmentation (DA) is an effective technique and is widely used in various deep learning tasks (including gaze estimation). Appearance-based gaze estimation aims to directly learn a mapping from face images to gaze directions. Since subtle changes in eye regions are important for gaze estimation, direct data augmentation on faces is likely to damage key features in the eye region. We propose a facial landmarks based region-level data augmentation method. The method use facial landmarks to divide the face into eye regions and non-eye regions. Then we generate face images under different data augmentation methods (illumination conditions, random occlusion and Gaussian blur) by augmenting non-eye regions. We preserve the key features of eye regions. And the features of non-eye regions are augmented. In this way, the proposed method can better utilize non-eye region features. We conduct experiments on both two popular datasets (GazeCapture, MPIIFaceGaze). Comprehensive experiments show that the proposed method achieves promising results on both two datasets. Wide range of gaze estimation based application will be aspired from this work.

Keywords Gaze estimation · Data augmentation · Facial landmarks · Human-computer interaction

1 Introduction

Gaze estimation is an important task and has widespread applications in many fields. For example, gaze estimation techniques are widely used in human attention diagnosis, especially fatigue driving [17], saliency detection [8,34,33], etc. Gaze has also became a newly developing human-computer interaction method[38,32, 37,24,25]. To enable such applications, accurate gaze estimation methods are important.

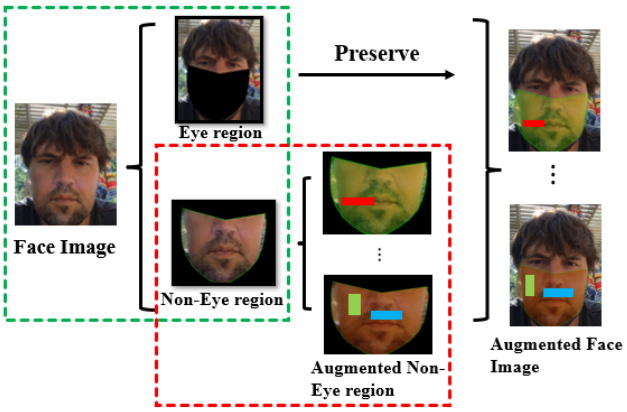

Fig. 1 Overview of the proposed method. The green box shows facial landmarks based region mask. Face images are divided into eye regions and non-eye regions. Eye regions are completely preserved. The red box shows non-eye regions data augmentation. Non-eye regions are augmented by generating different data augmentation methods. Finally, we get the augmented face images.

Up to now, many gaze estimation methods have been proposed including model-based and appearance-based methods. Model-based methods aim to fit a 3D eye model to the image and calculate gaze via specific geometric constrains [36]. They usually require some dedicate devices like high resolution cameras, RGB-D cameras [1,30]and infrared cameras [43]. Appearance-based methods do not require dedicated devices and use web cameras to capture human eye appearance and regress gaze from the appearance. Appearance-based methods which directly map gaze from appearance have made great progress. Some appearance-based methods

using convolutional neural networks (CNNs) [40,18,4, 14,2,41] have been proposed and show convincing results. This also benefits from the powerful feature extraction capability of CNN.

With the further research on gaze estimation, it is found that face images contain rich information[18]. Data augmentation has been proved to be an essential technique for increasing the effective data size and promoting the diversity of training examples[11,42]. We can use data augmentation to augment facial information[22]. Subtle changes in eye regions are important for the gaze estimation[3]. However, direct data augmentation on faces is likely to damage key features in the eye region. Besides, data augmentation methods such as flipping and affine transformation can reduce the performance of gaze estimation. Because these methods change the implicit features of face images such as head pose[40]. Gong et al.[11] propose a simple yet highly effective approach called KeepAugment. This method is first to use the saliency map to detect important regions on the original images and then preserve these informative regions during augmentation. This information preserving strategy allows us to generate more faithful training examples.

So, we propose a facial landmarks based region-level data augmentation method for gaze estimation. Firstly, we use landmarks to divide the face into eye regions and non-eye regions. Then, we generate face images under different data augmentation methods (illumination conditions, random occlusion and Gaussian blur) by augmenting non-eye regions. Finally, we fuse the augmented non-eye regions with eye regions to generate the final augmented image. As shown in Fig. 1, the proposed method can ensure the key features of eye regions are not affected and enrich the features of non-eye regions. As with most data augmentation methods, our region-level data augmentation method allows our model to show better generalization ability. The overall structure of gaze estimation is shown in Fig. 2.

In summary, the contributions of this paper are as follow:

1. We propose a facial landmarks based region-level data augmentation method for gaze estimation.

2. We comprehensively consider the influence of illumination conditions, random occlusions and Gaussian blur on images to generate the facial image in different data augmentation methods.

3. We conduct experiments on two popular datasets (GazeCapture, MPIIFaceGaze). On the GazeCapture dataset, the proposed method is close to the state of the art (SOTA) on tablets and outperforms the SOTA on mobile phones. In addition, the proposed method also outperforms SOTA methods on the MPIIFaceGaze dataset.

The rest of the paper is organized as following. Section 2 summarizes the overview of related works. Section 3 describes our data augmentation method. In Section 4, we first describe the experimental setup including datasets, training setup and evaluation metrics. Then, we compare the experimental results on two popular public dataset with recent SOTA methods and conduct ablation studies. In Section 5, we summarize our work.

2 Related works

2.1 Gaze estimation

As an active research topic, many different methods have been proposed to address gaze estimation problem. These methods can generally be categorized into model-based methods and appearance-based methods.

Model-based methods estimate gaze based on detecting geometric features such as contours, reflection and eye corners [12]. To acquire accurate eye location, distinct eye features like corneal reflection[23], pupil center[31] and iris contour[1,20] are widely used. Wen et al.[35] use convergence constraint which allows calibration without knowing exact gaze location and a person independent gaze corrector to reduce system error.

Appearance-based methods aim to find the direct mapping function from image appearance to gaze direction or gaze location. And these methods can be further categorised depending on whether the regression target is in 2D or 3D.

2D Gaze estimation assumed a fixed head pose of the target person[31] and consequently focused on the 2D gaze estimation task where the estimator is trained to output on screen gaze locations. Most previous works used a single eye image as input to the regressor and only few considered alternative methods such as using two images, one of each eye[16]. Krafka et al.[18] propose a multi-region 2D gaze estimation architecture — iTracker that takes individual eye images, the face image and a face grid as input. In particular, this work provides a large scale public dataset — GazeCapture. Junfeng et al. [14] present an on-device few-shot personalization method for 2D gaze estimation. The method reduces the number of calibration points required by the user. Guo et al.[13] propose a new tolerant and talented (TAT) training scheme, which is an iterative random knowledge distillation framework enhanced with cosine similarity pruning and aligned orthogonal initialization. Recently, Bao et al.[2] propose an accurate appearance-based gaze estimation method named AFF-Net. The

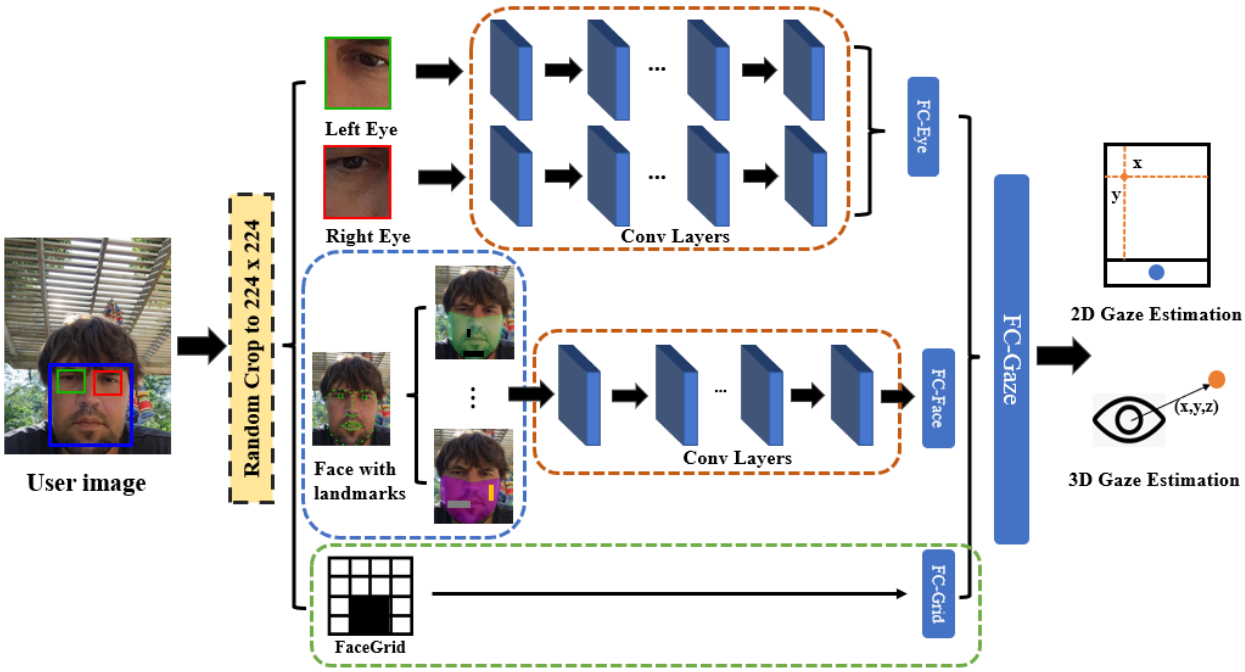

Fig. 2 Overview of the gaze estimation structure. Our structure contains three input branches, eye image input branch, face image input branch and FaceGrid input branch. Eye image input branch includes two inputs which respectively input left eye and right eye images (all of size 224 × 224). In particular, the green box represents FaceGrid input branch which comes from iTracker[18] and this part is only used on the GazeCapture dataset. The orange boxs represent the model we used. We use ResNet-50 as our model for gaze estimation. The blue box represents to generate different augmented images with the proposed method. Finally, all branches are connected through FC layers to output the gaze point or gaze direction.

proposed AFF-Net improves gaze tracking accuracy by adaptively fusing two eye features and face appearance characteristics guided eye feature extraction. And the AFF-Net achieved SOTA on the GazeCapture dataset at that time.

3D Gaze estimation is trained to output 3D gaze directions in the camera coordinate system[10,21]. To facilitate model training, Sugano et al.[29] propose a data normalization technique to restrict the appearance variation into a normalized training space. [39] present a method for in-the-wild appearance-based gaze estimation using multimodal convolutional neural networks. In addition, this work also provides a popular dataset—MPIIFaceGaze. Zhang et al.[40] propose a spatial weights CNN method that leveraged information from the face. This method demonstrate that facial images are more robust to facial appearance variation caused by extreme head pose and gaze directions as well as illumination. Fischer et al.[9] propose a novel approach called RT-GENE for ground truth gaze estimation in these natural settings. The method apply semantic image inpainting to the area covered by the glasses to bridge the gap between training and testing images by removing the obtrusiveness of the glasses. Cheng et al.[3] propose a plug-and-play domain-generalization framework that purifies gaze feature to improve the performance in unknown target domains without touching the domain. This method eliminates gaze irrelevant factors. The AFF-Net [2] which achieves SOTA performance on GazeCapture dataset can also achieve SOTA on MPIIFaceGaze dataset.

2.2 Data augmentation

Data augmentation is a widely used technique to artificially enlarge the training dataset from existing data using various transformations. Two classes of augmentation techniques are widely used for achieving SOTA results on computer vision tasks:

Image-Level Augmentation apply label invariant transformations on the whole image such as solarization, sharpness, posterization, color normalization and illumination[22]. Image-level transformations are often manually designed and heuristically chosen. Cubuk et al.[5] propose the AutoAugment which applies reinforcement learning to automatically search optimal compositions of transformations. Subsequent works including RandAugment[6] and Fast AutoAugment[19] reduce the heavy computational burden of searching on

the space of transformation policies by designing more compact search spaces.

Region-Level Augmentation transform a specific area of an image. Some classical image transformation methods are also widely used in model training[27, 15, 28] such as, translation, rotation, flipping, cropping, etc. These classic techniques are fundamental to obtain highly generalized deep models. It is shown in some literature that abandoning certain information in training images is also an effective approach to augment the training data. Cutout [7] and random erasing [42] work by randomly masking out or modifying rectangular regions of the input images and creating partially occluded data examples outside the span of the training data. This procedure could be conveniently formulated as applying randomly generated binary masks to the original inputs. However, both two methods may mask or even damage key features of eye regions. Recently, Gong et al.[11] first use saliency map to measure the importance of each region and propose to avoid cutting important regions for region-level data augmentation methods. As mentioned earlier, subtle changes in the eye region directly affect the performance of gaze estimation. Direct data augmentation on faces is likely to damage key features in the eye region. Therefore, it is important to select an appropriate region for data augmentation.

In this work, we propose a region-level data augmentation method for gaze estimation. We can generate face images under different data augmentation methods (illumination conditions, random occlusion and Gaussian blur) by augmenting non-eye regions.

3 Proposed method

As shown in Fig. 1, the proposed method consists of two steps—facial landmarks based region mask and non-eye regions data augmentation. We use facial landmarks to divide the face into eye regions and non-eye regions. For eye regions, we completely preserve the features of the eye region. For non-eye regions, we generate the non-eye region under different illumination conditions by adjusting brightness, contrast and stauration. Besides, we randomly erase parts of the non-eye region to generate random occlusion in different data augmentation methods. Finally, we fuse the augmented non-eye region with the eye region to generate the augmented face image. The following is a detailed description of two steps.

3.1 Facial landmarks based region mask

Firstly, as shown in Fig. 3 , we extract the 2D facial landmarks with the Dlib toolkit. Landmark is a method of facial feature point extraction. This method uses the ERT (Ensemble of regression trees) cascaded regression algorithm and uses a series of calibrated face images for training and then generates a model. When we input a face image, the algorithm will generate an initial shape and use this to estimate the approximate location of the face feature points. Then, the gradient boosting algorithm is used to reduce the sum of squared errors of the initial shape and ground truth. Least squares are used to minimize the error and resulting in cascaded regression factors for each level.

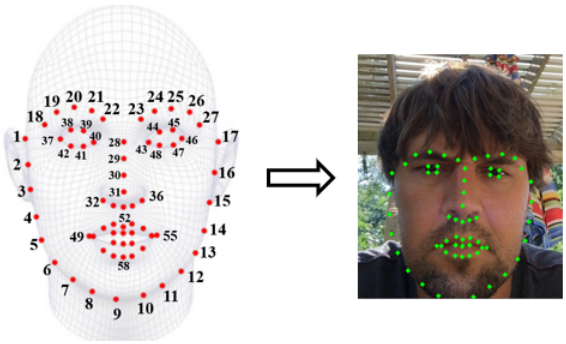

Fig. 3 The illustration of 68 facial landmarks. These facial landmarks are used to mark keypoints in face images. 68 facial landmark keypoints are defined in iBUG 300-W dataset [26]

To explain the method more theoretically, we introduce some notations. Let $X_i \in \mathbb{R}^2$ be the x, y coordinates of the ith facial landmark in an image I. Then the vector $\mathbf{Z} = \left(X_1^T, X_2^T, ..., X_i^T\right) \in \mathbb{R}^{2p}$ denotes the coordinates of all the i facial landmarks in I. We refer to the vector $\mathbf{Z}$ as the shape. We use $\hat{\mathbf{Z}}^{(t)}$ to denote our current estimate of $\mathbf{Z}$. Each regressor, $r_t(.,.)$, in the cascade predicts an update vector from the image and $\hat{\mathbf{Z}}^{(t)}$ that is added to the current shape estimate $\hat{\mathbf{Z}}^{(t)}$ to improve the estimate:

$$\hat{\mathbf{Z}}^{(t+1)} = \hat{\mathbf{Z}}^{(t)} + r_t\left(I, \hat{\mathbf{Z}}^{(t)}\right) \tag{1}$$

The keypoint of the cascade is that the regressor r_t makes its predictions based on features such as pixel intensity values computed from I and indexed relative to the current shape estimate $\hat{\mathbf{Z}}^{(t)}$.

When we get a face image with 68-point calibration, we connect the 1-17 calibration points in turn to draw the face contour. Then, we connect the No. 1 calibration point, the No. 30 calibration point and the No. 17 calibration point in turn to generate masks for non-eye

regions. Finally, we obtain the eye region and non-eye region images using generated masks, as shown in Fig. 4.

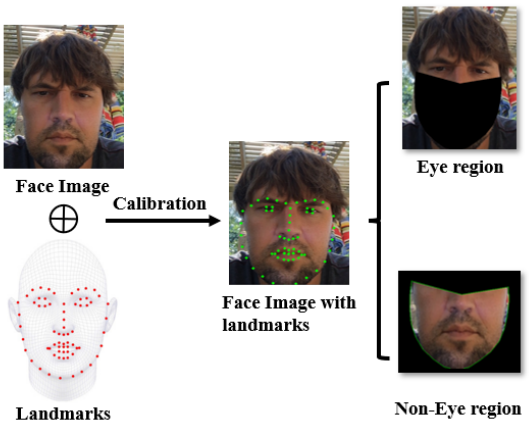

Fig. 4 The process of dividing face image into eye regions and non-eye regions. We first use landmarks to generate masks for non-eye regions. Then, the face image is divided into eye regions and non-eye regions using masks.

3.2 Non-eye regions data augmentation

This section describes the data augmentation method for non-eye regions, as shown in Fig. 5, including illumination conditions, random occlusion and Gaussian blur.

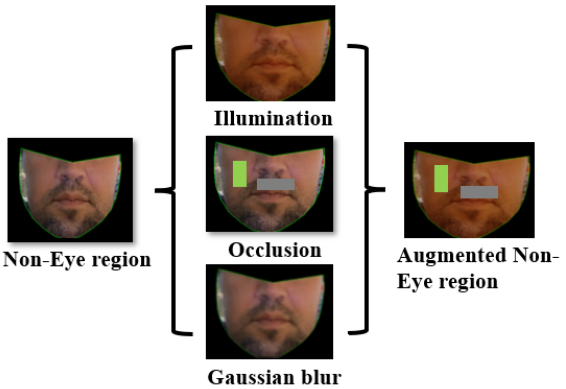

Fig. 5 The augmentation process of the non-eye region. We first generate non-eye regions under illumination conditions, random occlusion and Gaussian blur. Then, we fuse these three data augmentation methods to generate augmented non-eye region images.

Random Erasing In training, we randomly erase with a certain probability. In this process, non-eye region images with various levels of occlusion are generated. For a non-eye region image I in a mini-batch, the probability of it undergoing random erasing is p and the probability of it being kept unchanged is 1-p. Firstly, this method randomly selects a rectangle region $I_e \subseteq I$ in the non-eye region and erases its pixels with random values. Assume that the size of the non-eye region is $W \times H$. W and H refer to the width and height of the farthest border of the non-eye region. The area of the image is:

$$S = W \times H \tag{2}$$

Then, we randomly initialize the area of erasing rectangle region to S_e, where $\frac{S_e}{S}$ is in range specified by minimum S_l and maximum S_h. The aspect ratio of erasing rectangle region is randomly initialized between r_1 and r_2. We set it to r_e. The size of I_e is:

$$W_e = \sqrt{\frac{S_e}{r_e}} \qquad H_e = \sqrt{S_e \times r_e} \tag{3}$$

Finally, this method randomly initialize a point $p^* = (x_e, y_e)$ in I. If $x_e + W_e \leq W$ and $y_e + H_e \leq H$, the selected rectangle region is:

$$I_e = (x_e, y_e, x_e + W_e, y_e + H_e) \tag{4}$$

Otherwise repeat the above process until an appropriate I_e is selected. And erased regions are filled with random color patches.

Gaussian Blur We found that there are some detailed features in the non-eye regions such as beards, wrinkles, etc. We use Gaussian blur to reduce the influence of these gaze irrelevant detailed features on gaze estimation. Gaussian blur is an image processing method based on Gaussian function, i.e.,

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \tag{5}$$

where μ is the mean of x and σ is the standard deviation of x.

We take each current calculation point as the origin. So $\mu = 0$. And the Eq. (5) is further simplified to:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/2\sigma^2} \tag{6}$$

In the normal distribution curve, the closer to the "center point", the larger the value. Otherwise, the smaller the value. So, we just need to take the "center point" as the origin and assign weights to other points according to their positions on the normal curve. And then a weighted average can be obtained. That is:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{7}$$

This achieves the effect of blurring the image. In this way, we can reduce the influence of gaze irrelevant features such as beards, wrinkles, etc., in non-eye regions on the gaze estimation.

Simulate Illumination Conditions Illumination is one of the important factors that affect the facial features. Due to the changes in illumination conditions, the same face appears differently[22]. In our illumination synthesis process, as shown in Fig.6, we generate face images by adjusting brightness, contrast and stauration.

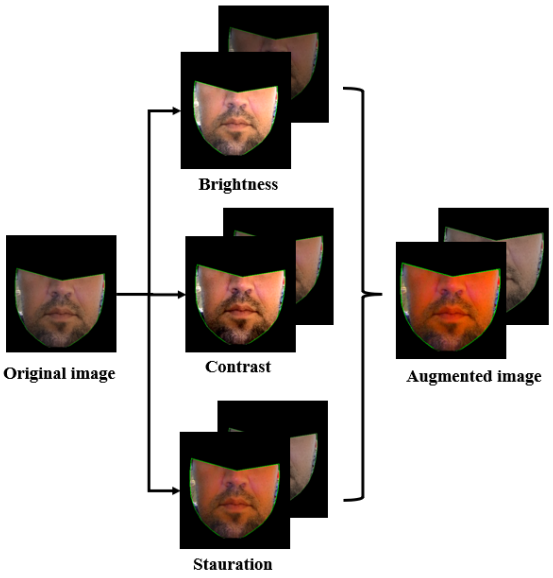

Fig. 6 Non-eye region images under different illumination conditions. We use brightness, contrast and stauration to generate augmented images under different illumination conditions

Finally, as shown in Fig. 7, we compare images augmented directly on the face and images augmented with the proposed method.

If we augment the face image directly, it is likely to damage the key features of eye regions such as the eye regions are erased. Hence, our data augmentation method only works on the non-eye region and the key features of the eye region are fully preserved. The proposed method can effectively augment facial data and reduce the influence of gaze irrelevant features on gaze estimation.

4 Experiments

4.1 Experimental setup

A. Datasets

We conduct experiments in two famous dataset: GazeCapture[18] and MPIIFaceGaze[39].

GazeCapture is the largest 2D gaze dataset with more than 2M images from more than 1,400 subjects. The dataset is captured by mobile phones or tablets in different orientations. There are 1,490,959 frames have both face and eye detections, which are further divided into 1,251,983 training images, 59,480 validation images and 179,496 test images. For training, each of the samples is treated independently while for testing. We average the predictions of the samples to obtain the prediction on the original test sample.

MPIIFaceGaze contains 213,659 images from 15 laptop cameras. The dataset is manually calibrated with 6-point landmarks and pupil data for 37,667 faces. It is a widely used dataset for gaze estimation problem. The dataset collected by laptops has a larger prediction space than GazeCapture.

B. Training details

The experiments are implemented using PyTorch. We use the ResNet-50 as experimental model. Network parameters are initialized by the default initialization of PyTorch. The model is trained for 18 epochs on the GazeCapture and MPIIFaceGaze dataset. The batch size is set to 32. The initial learning rate is set to 0.001 and it becomes 0.0001 after 8 epochs. The optimizer uses SGD. Further, we use a momentum of 0.9 and a weight decay of 0.0005 throughout the training procedure.

C. Evaluation metrics

For gaze point estimation on GazeCapture dataset, we use the Euclidean distance to measure the error between the predicted and the ground truth gaze points, i.e.,

$$d_e = \frac{1}{M}\sum_{i}^{M}\left\|p^i - \hat{p}^i\right\|_2 \tag{8}$$

where M is the total number of images in GazeCapture. p^i and $\hat{p}^i$ are the ground truth and the predicted gaze point, for the ith image.

For gaze direction estimation on the MPIIFaceGaze dataset, the angle deviation between the estimation and the ground truth gaze direction is used for the performance evaluation, i.e.,

$$a_e = \frac{1}{N}\sum_{i}^{N} arccos\frac{\left\langle a^i, \hat{a}^i\right\rangle}{|a^i|\,|\hat{a}^i|} \tag{9}$$

where N is the total number of images in the MPIIFaceGaze dataset. a^i is the ground truth gaze direction of the ith image and its prediction is $\hat{a}^i$. $\left\langle a^i, \hat{a}^i\right\rangle$ refers to the inner product between a^i and $\hat{a}^i$.

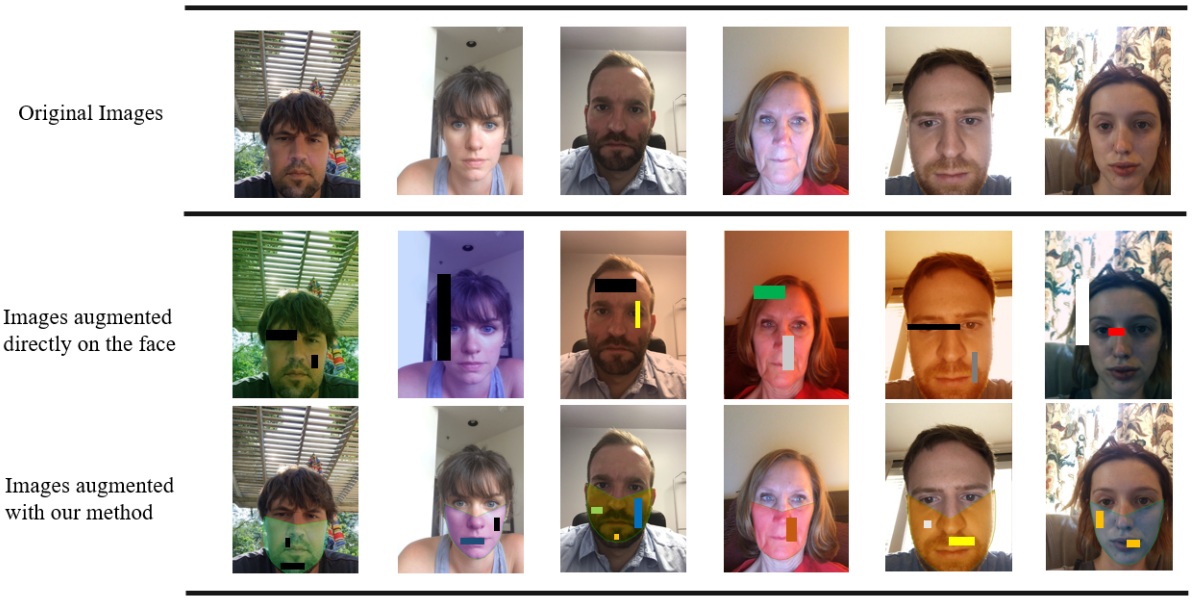

Fig. 7 Comparison of two data augmentation methods. The 1st row shows original face images and the 2nd row shows images augmented directly on the face. The 3rd row shows images augmented with the proposed method.

4.2 Performance comparison

We conduct two experiments test different methods on the GazeCapture and MPIIFaceGaze dataset. The results are shown in Tables 1 and 2.

The GazeCapture dataset is almost the largest gaze dataset in mobile devices. We first conduct performance evaluation of our method on the GazeCapture dataset. We choose four methods for comparison on GazeCapture, which are iTracker[18], SAGE[14], TAT[13] and AFF-Net[2]. To the best of our knowledge, AFF-Net shows the SOTA performance on GazeCapture.

Table 1 GAZE ESTIMATION ERROR IN CENTIMETERS COMPARES WITH SOTA METHODS ON THE GAZECAPTURE DATASET.

Methods	Phone error(cm)	Tablet error(cm)
iTracker[18]	1.86	2.81
SAGE[14]	1.78	2.72
TAT[13]	1.77	2.66
AFF-Net[2]	1.62	2.3
Proposed method	**1.58**	2.36

The proposed method achieves 1.58 cm error on mobile phones and 2.36 cm error on tablets which outperforms SOTA methods on mobile phones. For mobile phone test, the iTracker has the highest error as 1.86 cm. SAGE and TAT have similar performance around 1.77 cm, improve about 5% from iTracker. AFF-Net achieves 1.62 cm error which outperforms both two methods significantly. The proposed method achieves 1.58 cm that outperforms AFF-Net and improves about 15% from iTracker. For the more challenging tablet test, the error of iTracker is 2.81 cm. SAGE and TAT has similar performance around 2.69 cm. AFF-Net achieves 2.3 cm error which is almost 0.39 cm lower than SAGE and TAT. The proposed method achieves similar results to AFF-Net and the error is 2.36 cm which is 12.3% lower than SAGE and TAT. The proposed method can improve the performance of gaze estimation on tablets than most previous methods. These experiment results show that the proposed method has clear advantages over other methods, espeically on mobile phone.

To further demonstrate the advantage of our method, we conduct more experiments on the MPIIFaceGaze dataset. We choose iTracker, Spacial Weights CNN[40], RT-GENE[9] and AFF-Net as the compared methods. Because most of them show outstanding performance in the 3D gaze position estimation task on MPIIFaceGaze. We list the result in Table 2.

The iTracker has the highest error as 6.2 degree angular error. The Spatial weights CNN and RT-GENE have similar performance around 4.8 degree angular error, improve about 22.6% from iTracker. The AFF-Net almost shows SOTA performance in 3D gaze direction estimation task on MPIIFaceGaze. AFF-Net achieves 4.4 degree angular error, improve about 8% from Spa-

Table 2 GAZE ESTIMATION ERROR IN DEGREES COMPARES WITH SOTA METHODS ON THE MPIIFaceGaze DATASET.

Methods	3D gaze direction error(degree)
iTracker[18]	6.2
Spatial weights CNN[40]	4.8
RT-GENE[9]	4.8
AFF-Net[2]	4.4
Proposed method	**3.95**

tial weights CNN and RT-GENE. As shown in Table 2, the proposed method achieves the performance of 3.95 degree angular error which significant performs better than other SOTA methods. Compared with AFF-Net, the proposed method achieves a performance improvement of 10.2%. Note that, MPIIFaceGaze dataset is collected from laptop. These result demonstrate that our method also can perform well in the laptop. Further, both two experiments prove the generality and effectiveness of the proposed method on both datasets.

4.3 Ablation studies

In this section, we conduct two ablation experiments to demonstrate the effectiveness of the facial landmarks based region mask and non-eye regions data augmentation on the GazeCapture dataset. We still use ResNet-50 as the training model.

A. Full face v.s. non-eye region DA comparison

First, we test the error of the model without using data augmentation. Then, we directly augment faces and test error. We compare both two experimental results with the result obtained by training with the proposed method. The results are shown in Table 3.

Table 3 Experimental results for facial landmarks based region mask.

Full face	Non-Eye region	Phone error(cm)	Tablet error(cm)
×	×	1.68	2.59
✓	×	1.73 ↑	2.82 ↑
×	✓	**1.58**	**2.36**

The results are shown in Table 3. Without data augmentation, the error of mobile phones and tablets is 1.68 cm and 2.59 cm. Data augmentation methods on full face reduce the performance of the model. The error of mobile phone and tablet is 1.73 cm and 2.82 cm which increase by 0.05 cm and 0.23 cm. Because data augmentation methods apply directly on full face (including the eye region) reducing the availability of key features in eye regions. For example, the eye region may be completely erased, as shown in Fig7 (row 2). With non-eye regions data augmentation, the error of mobile phone and tablet is 1.58cm and 2.36cm. The error is reduced by 6% and 9% for mobile phones and tablets.

B. Different DA methods comparison for non-eye regions

We study the influence of different data augmentation methods on non-eye regions. We use five data augmentation methods such as illumination, occlusion, Gaussian blur, flip and affine transformation. First, we test the error without data augmentation. Then, we study two other data augmentation methods (flip and affine transformation). Finally, we test the error when using illumination, occlusion and Gaussian blur to augment non-eye regions. In this experiment, we use the overall error for phones and tablets to represent the performance of our model on GazeCapture. The results are shown in Table 4.

Table 4 Experimental results for different data augmentation methods on non-eye regions

Illumination	Occlusion	Gaussian blur	Flip	Affine transformation	Error (cm)
×	×	×	×	×	2.1
×	×	×	✓	×	2.39↑
×	×	×	×	✓	2.47↑
✓	×	×	×	×	1.93
×	✓	×	×	×	1.96
×	×	✓	×	×	1.98
✓	✓	×	×	×	1.89
✓	×	✓	×	×	1.91
×	✓	✓	×	×	1.86
✓	✓	✓	×	×	**1.83**

As shown in Table 4, flip and affine transformation are not suitable for gaze estimation. Their errors increase to 2.39 and 2.47 cm. Because both two methods change the implicit information of the face, such as head pose, etc. When non-eye regions augmented with one or two data augmentation methods (illumination, occlusion or Gausman blur), the errors range from 1.86 cm to 1.98 cm and the average error is 1.92 cm. Compared to the result without data augmentation, the error is re-

duced by 8%. In particular, when we augment non-eye regions with three methods(illumination, occlusion and Gaussian blur), the error is 1.83 cm which is almost reduced by 13%. These experimental results demonstrate that the proposed data augmentation method is effective.

5 Conclusion

In this work, we propose a facial landmarks based region-level data augmentation method for gaze estimation. The face are divided into eye regions and non-eye regions. Then, we generate face images under different data augmentation methods by augmenting non-eye regions. The proposed method achieves excellent performance on the largest 2D gaze dataset — GazeCapture dataset. On tablets the proposed method is close to the current SOTA methods and on mobile phones outperforms the SOTA methods. In particular, the proposed method also outperforms the SOTA methods on the MPIIFaceGaze dataset. These results prove the generality and effectiveness of the proposed method on both two popular datasets. Gaze estimation and its applications are promising research topics. We believe researchers will be inspired from this work.

References

1. Alberto Funes Mora, K., Odobez, J.M.: Geometric generative gaze estimation (G3E) for remote RGB-D cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1773–1780 (2014)
2. Bao, Y., Cheng, Y., Liu, Y., Lu, F.: Adaptive feature fusion network for gaze tracking in mobile tablets. In: 25th International Conference on Pattern Recognition, pp. 9936–9943. IEEE (2021)
3. Cheng, Y., Bao, Y., Lu, F.: PureGaze: Purifying gaze feature for generalizable gaze estimation. CoRR **abs/2103.13173** (2021)
4. Cheng, Y., Zhang, X., Lu, F., Sato, Y.: Gaze estimation by exploring two-eye asymmetry. IEEE Transactions on Image Processing **29**, 5259–5272 (2020)
5. Cubuk, D.E., Zoph, B., Mané, D., Vasudevan, V., Le, V.Q.: Autoaugment: Learning augmentation policies from data. CoRR **abs/1805.09501** (2018)
6. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 702–703 (2020)
7. Devries, T., Taylor, W.G.: Improved regularization of convolutional neural networks with cutout. CoRR **abs/1708.04552** (2017)
8. Fan, D.P., Cheng, M.M., Liu, J.J., Gao, S.H., Hou, Q., Borji, A.: Salient objects in clutter: Bringing salient object detection to the foreground. In: Proceedings of the European Conference on Computer Vision, pp. 186–202 (2018)
9. Fischer, T., Chang, H.J., Demiris, Y.: RT-GENE: Real-time eye gaze estimation in natural environments. In: Proceedings of the European Conference on Computer Vision, pp. 334–352 (2018)
10. Funes-Mora, K.A., Odobez, J.M.: Gaze estimation in the 3D space using RGB-D sensors. International Journal of Computer Vision **118**(2), 194–216 (2016)
11. Gong, C., Wang, D., Li, M., Chandra, V., Liu, Q.: Keepaugment: A simple information-preserving data augmentation approach. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1055–1064 (2021)
12. Guestrin, E.D., Eizenman, M.: General theory of remote gaze estimation using the pupil center and corneal reflections. IEEE Transactions on Biomedical Engineering **53**(6), 1124–1133 (2006)
13. Guo, T., Liu, Y., Zhang, H., Liu, X., Kwak, Y., In Yoo, B., Han, J.J., Choi, C.: A generalized and robust method towards practical gaze estimation on smart phone. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, pp. 1131–1139 (2019)
14. He, J., Pham, K., Valliappan, N., Xu, P., Roberts, C., Lagun, D., Navalpakkam, V.: On-device few-shot personalization for real-time gaze estimation. IEEE/CVF International conference on computer vision Workshops pp. 1149–1158 (2019)
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
16. Huang, Q., Veeraraghavan, A., Sabharwal, A.: TabletGaze: Dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. Machine Vision and Applications **28**(5), 445–461 (2017)
17. Ji, Q., Yang, X.: Real-time eye, gaze, and face pose tracking for monitoring driver vigilance. Real Time Imaging **8**, 357–377 (2002)
18. Krafka, K., Khosla, A., Kellnhofer, P., Kannan, H., Bhandarkar, S., Matusik, W., Torralba, A.: Eye tracking for everyone. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2176–2184 (2016)
19. Lim, S., Kim, I., Kim, T., Kim, C., Kim, S.: Fast autoaugment. Annual Conference on Neural Information Processing Systems pp. 6662–6672 (2019)
20. Lu, F., Gao, Y., Chen, X.: Estimating 3D gaze directions using unlabeled eye images via synthetic iris appearance fitting. IEEE Transactions on Multimedia **18**(9), 1772–1782 (2016)
21. Lu, F., Sugano, Y., Okabe, T., Sato, Y.: Gaze estimation from eye appearance: A head pose-free method via eye image synthesis. IEEE Transactions on Image Processing **24**(11), 3680–3693 (2015)
22. Lv, J.J., Shao, X.H., Huang, J.S., Zhou, X.D., Zhou, X.: Data augmentation for face recognition. Neurocomputing **230**, 184–196 (2017)
23. Nakazawa, A., Nitschke, C.: Point of gaze estimation through corneal surface reflection in an active illumination environment. In: European Conference on Computer Vision, pp. 159–172. Springer (2012)
24. Piumsomboon, T., Lee, G., Lindeman, R.W., Billinghurst, M.: Exploring natural eye-gaze-based interaction for immersive virtual reality. In: IEEE Symposium on 3D User Interfaces, pp. 36–39. IEEE (2017)
25. Ren, L., Huang, H., Wang, H., Yang, Z.: Gazegrid: A novel interaction method based on gaze estimation. In:

IEEE International Conference on Automatic Face and Gesture Recognition, pp. 1–5. IEEE (2021)
26. Sagonas, C., Antonakos, E., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: database and results. Image and Vision Computing **47**, 3–18 (2016)
27. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
28. Srivastava, K.R., Greff, K., Schmidhuber, J.: Training very deep networks. Annual Conference on Neural Information Processing Systems pp. 2377–2385 (2015)
29. Sugano, Y., Matsushita, Y., Sato, Y.: Learning-by-synthesis for appearance-based 3D gaze estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1821–1828 (2014)
30. Sun, L., Liu, Z., Sun, M.T.: Real time gaze estimation with a consumer depth camera. Information Sciences **320**, 346–360 (2015)
31. Valenti, R., Sebe, N., Gevers, T.: Combining head pose and eye location information for gaze estimation. IEEE Transactions on Image Processing **21**(2), 802–815 (2011)
32. Wang, H., Dong, X., Chen, Z., Shi, B.E.: Hybrid Gaze/EEG brain computer interface for robot arm control on a pick and place task. In: 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 1476–1479. IEEE (2015)
33. Wang, W., Shen, J., Dong, X., Borji, A., Yang, R.: Inferring salient objects from human fixations. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(8), 1913–1927 (2019)
34. Wang, W., Shen, J., Yang, R., Porikli, F.: Saliency-aware video object segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(1), 20–33 (2017)
35. Wen, Q., Bradley, D., Beeler, T., Park, S., Hilliges, O., Yong, J., Xu, F.: Accurate real-time 3D gaze tracking using a lightweight eyeball calibration. In: Computer Graphics Forum, vol. 39, pp. 475–485. Wiley Online Library (2020)
36. Xiong, X., Liu, Z., Cai, Q., Zhang, Z.: Eye gaze tracking using an RGBD camera: A comparison with a rgb solution. In: Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, pp. 1113–1121 (2014)
37. Zhang, X., Sugano, Y., Bulling, A.: Everyday eye contact detection using unsupervised gaze target discovery. In: Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology, pp. 193–203 (2017)
38. Zhang, X., Sugano, Y., Bulling, A.: Evaluation of appearance-based methods and implications for gaze-based applications. In: Proceedings of the Conference on Human Factors in Computing Systems, pp. 1–13 (2019)
39. Zhang, X., Sugano, Y., Fritz, M., Bulling, A.: Appearance-based gaze estimation in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4511–4520 (2015)
40. Zhang, X., Sugano, Y., Fritz, M., Bulling, A.: It's written all over your face: Full-face appearance-based gaze estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 51–60 (2017)
41. Zhang, Z., Lian, D., Gao, S.: RGB-D-based gaze point estimation via multi-column CNNs and facial landmarks global optimization. The Visual Computer **37**(7), 1731–1741 (2021)
42. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. In: Proceedings of the AAAI conference on artificial intelligence, vol. 34, pp. 13,001–13,008 (2020)
43. Zhu, Z., Ji, Q.: Novel eye gaze tracking techniques under natural head movement. IEEE Transactions on Biomedical Engineering **54**(12), 2246–2260 (2007)