Gaze Estimation with an Ensemble of Four Architectures

Xin Cai[1], Boyu Chen[2], Jiabei Zeng[1], Jiajun Zhang[2], Yunjia Sun[1],
Xiao Wang[2], Zhilong Ji[2], Xiao Liu[2], Xilin Chen[1], Shiguang Shan[1]
[1] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[2] Tomorrow Advancing Life (TAL) Education Group
{caixin20s, jiabei.zeng, sunyunjia18z, xlchen, sgshan}@ict.ac.cn
{chenboyu,zhangjiajun1, wangxiao15, jizhilong, liuxiao15}@tal.com

Abstract

This paper presents a method for gaze estimation according to face images. We train several gaze estimators adopting four different network architectures, including an architecture designed for gaze estimation (i.e.,iTracker-MHSA) and three originally designed for general computer vision tasks(i.e., BoTNet, HRNet, ResNeSt). Then, we select the best six estimators and ensemble their predictions through a linear combination. The method ranks the first on the leader-board of ETH-XGaze Competition, achieving an average angular error of 3.11° *on the ETH-XGaze test set.*

1. Introduction

Estimating people's eye-gaze according to facial images plays a fundamental role in varied applications of human-computer interaction[11, 16, 15], affective computing[7], and medical diagnosis[22, 12]. Though the gaze estimation task can be efficiently solved with deep learning-based approaches, we lack an appropriate and unified metric to compare different state-of-the-art methods. Additionally, existing gaze estimation datasets have limitations on the head pose and gaze variations and imperfect quality of images and ground truth labels. To advance the development of gaze estimation research, Zhang[25] proposed ETH-XGaze, a new gaze estimation dataset including more than one million high-resolution images of varying gaze and head poses.

Considering a larger variety of settings in ETH-XGaze, including variation of viewpoint, extreme gaze angles, lighting variation, and occluders like glasses, it is a challenge to conduct gaze estimation accurately on it. Basic neural models like Resnet-50 have limitation on providing accurate enough in a varying environment, and more delicate network architectures need to be explored for higher accuracy.

In this paper, we propose a gaze estimation network based on iTracker[14] and three other architectures for general computer vision purposes. We explore the utility of muti-scale, split-attention networks, and different training techniques in gaze estimation tasks.

2. Relate Work

Gaze estimation methods can be classified into model-based and appearance-based methods. Model-based methods generally use a geometric eye model with parameters inferred from localized eye landmarks such as the pupil center[8] and the iris edge[23, 1] to obtain gaze estimation result. Model-based methods usually require the specific device for high- resolution images and are limited by light conditions and short working distance, which is not suitable for gaze estimation under varying environments. Meanwhile, appearance-based methods directly estimate the gaze direction from the face images. Appearance-based methods[14, 27, 4] mostly learn a mapping from face or eyes images to gaze and rely on deep learning to extract feature from images. Appearance-based methods can estimate gazes despite light or appearance variations with the help of enough training data.

Based on previous works, different methods for learning based gaze estimation have been proposed. Cheng[2] proposed to use dilated convolutions and Cheng[3] came up with a coarse-to-fine network to improve gaze estimation result. Chen and Zhang [5] proposed to improve gaze estimation by exploring two-eye asymmetry.

3. Methods

The gaze estimation task is formulated as a regression from the normalized face images to the pitch-yaw gaze direction vectors. In ETH-XGaze Challenge, we adopted four different network architectures to train the gaze estimators, including an architecture designed for gaze estimation (i.e., iTracker-MHSA) and three originally designed for general

computer vision tasks (i.e., BoTNet, HRNet, ResNeSt). Then, we select the best six estimators and ensemble their predictions through a linear combination. Below, we introduce details of the adopted four network architectures.

3.1. iTracker-MHSA

To solve gaze point estimation problem, Krafka et al. [14] proposed iTracker, which combines the information from left and right eye images, face images along with face grid information. The face grid indicates the position of the face region in the captured image for gaze point estimation. We make some improvements on iTracker for our gaze direction estimation task and the key modifications in terms of input and architecture compared to iTracker are summarized as follows.

Figure. 1 illustrates the architecture of iTracker-MHSA. Firstly we remove the face grid branch since our gaze direction labels have been normalized in a norm camera space. Then we substitute eye backbone of iTracker for dilated Resnet50[10] and face backbone for convolution layers of Resnet50 to obtain better features. According to Chen[5], compared with canonical convolutions, dilated convolutions achieve remarkable accuracy gains on gaze estimation tasks. Last, to learn the relationship of face and eye features and adjust weights of face and eye features automatically, we use a multi-head self-attention mechanism in our designed network iTracker-MHSA. Specifically, we add a transformer encoder[20] to encode face and eye features, and then we use encoded features to estimate gaze.

The sub-network structure of the transformer encoder is shown in Fig.2. A transformer encoder has two sub-layers. The first is a multi-head self-attention layer, and the second is a fully connected feed-forward network. We implement a residual connection around each of the two sub-layers and use layer normalization to normalize the sum. As a result, the output of each sub-layer is LayerNorm(x + Sublayer(x)), where Sublayer is a multi-head self-attention layer or feed-forward layer. The detail of the multi-head self-attention mechanism can be found in [20]. Given face and eye features extracted by backbone as input, the encoder output encoded features with the same shape of input.

To locate eye positions in given face images, we use hrnet_w18[21] to detect facial landmarks. Since detected facial landmarks are decimals, we use RoI align [9] to crop eye images for accurate eye location.

During training iTracker-MHSA, online hard example mining strategy [17] are conducted to ensure models' ability to handle hard examples. Specifically, we sort losses of samples in a batch in descending order and double the losses of the top 30% samples.

3.2. BoTNet

A lot of network architectures have been proposed to extract images feature and improve downstream tasks such as image classification, object detection, and so on. Some of them are proved to be effective in our gaze estimation task.

BoTNet[18] proposed a method to replace spatial convolution layers with the multi-head self-attention layer proposed in the Transformer[20], which helps network learning global features of the input. Fig.3 shows how to change Resnet bottleneck block to a bottleneck transformer block. The structure of the multi-head self-attention layer is described in [18]. We design a network based on BoTNet as shown in Fig.4 We down-sample the input feature map using convolution with stride 2 and max pooling and then obtain a 2048-dimensional feature by three resnet bottleneck blocks and three bottleneck transformer blocks, followed by two fully connected layers to finish gaze estimation.

3.3. HRNet

High-resolution representation learning for eye plays an important role in gaze estimation. Different from other networks which downsampling feature maps, HRNet[19] develops a creative method to maintain the high-resolution representation of input through the whole inference process of model. HRNet connects high-to-low resolution convolutions in parallel and repeatedly implements fusions across parallel convolutions for high-resolution representations.

We conduct HRNetV2-W64[19] for learning a 1000-dim high-resolution representation of input face images. Based HRNet backbone, we add two fully connection layers (1000-128-2) to learning a mapping from the 1000-dim representation to the 2-dim gaze result.

3.4. ResNeSt

ResNeSt[24] is a network architecture that combines channel-wise attention with multi-path representation strategies to improve the network representation. A block of ResNeSt perfroms a series of transformations on low dimensional features and concatenates their outputs as in a multi-path network. Each transformation conducts a channel-wise attention strategy to capture the relationship of different feature maps.

Like HRNet method, we use ResNeSt269[24] as our backbone to extract features and employ three fully connection layers (2048-128-128-2) following the backbone to estimate gaze direction.

4. Experiments

We train the gaze estimators on ETH-XGaze dataset using the above four architectures with different settings. Then we choose six estimators that performed the best and ensemble their predictions to get the final prediction.

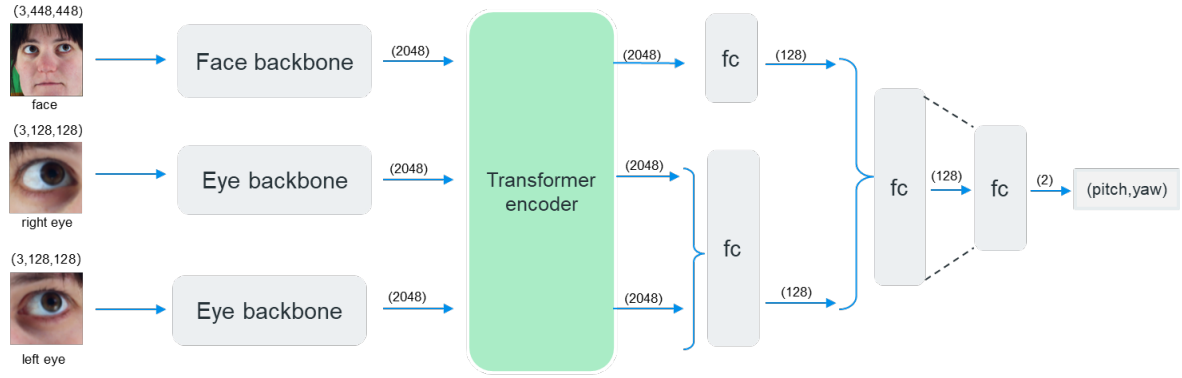

Figure 1. An overview of iTracker-MHSA architecture. The face backbone is Resnet50 and the eye backbone is dilated Resnet50.

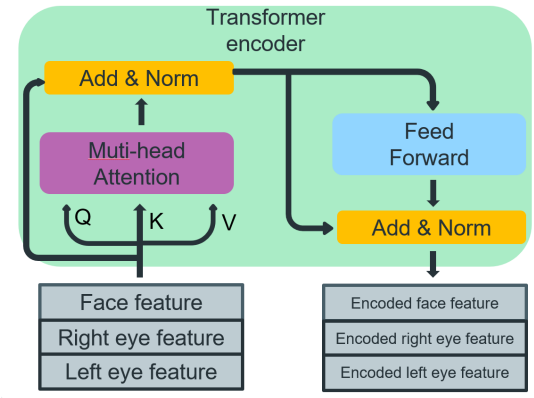

Figure 2. The structure of transformer encoder. The face backbone is Resnet50 and eye backbone is dilated Resnet50.Shapes of features is labelled in round brackets.

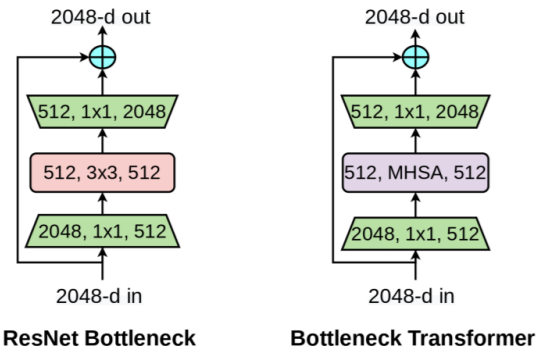

Figure 3. Left: A ResNet Bottleneck Block, Right: A Bottleneck Transformer block. The difference is the replacement of the spatial 3 × 3 convolution layer with Multi-Head Self-Attention.[18]

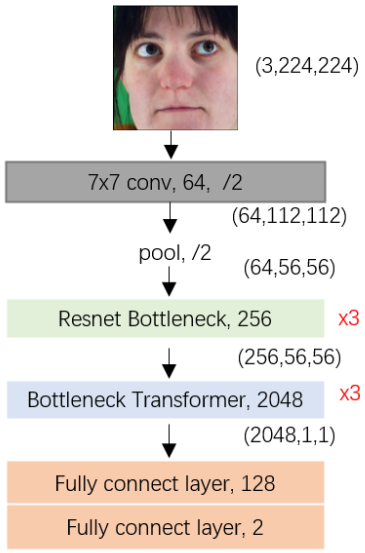

Figure 4. An overview of our boTNet based network. The shapes of features are labelled in round brackets.

4.1. Data pre-processing

Image size. We directly used the challenge-provided 224×224 facial images which are normalized by [26]. We resize the input frame to different sizes (e.g., 448×448, 640 × 640) by bilinear interpolation when training and test models, and experiments show changing image size is vital to gaze estimation.

Data augmentation. We flip training images horizontally to augment data. The yaw labels of flipped data are opposites of original labels and the pitch labels are changeless.

Validation set split. We choose 10 subjects as a validation set from the origin training set by sex, skin tone, and whether to wear glasses. The subject ids of validation set are: 03, 32, 33, 48, 52, 62, 80, 88, 101, 109 and some ex-

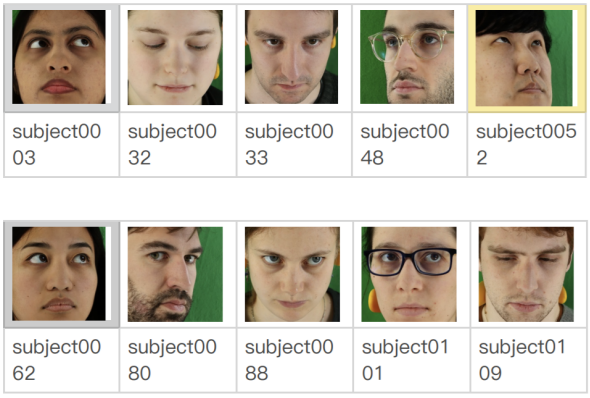

Figure 5. Examples of our validation set.

amples are shown in Fig 5. We train all models only using training set and choose best models on the validation set to conduct inference on test set in our experiments.

4.2. Implementation details

We train the models on 8 NVIDIA V100 GPUs using DistributedDataParallel(DDP) Pyorch and synchronized batch normalization strategy. All models we use are pretrained on Imagenet[6]. We use L1 loss function, exponential learning rate schedule strategy and adam[13] optimizer to train models.

For **iTracker-MHSA** with input size 448, the batch size is (30 $\times$ 8), the initial learning rate is (1e-4 $\times$ 8) . We choose the best performed model on the validation set among different training epochs.

For **BoTNet** with input size 224, the batch size is (24 $\times$ 8) and the initial learning rate is (1e-4 $\times$ 8).

For **ResNeST** with input size 448, the batch size is (12 $\times$ 8), the initial learning rate is (1e-4 $\times$ 8), and the chosen epoch is 10.

As for **HRNet**, three models are selected. We use horizontal flip to augment the training set and step learning rate schedule strategy when training them.

For HRNet with input size 640, the batch size is (24 $\times$ 8) and the initial learning rate is (2.5e-5 $\times$ 8). And for HRNet with input size 768, the batch size is (16 $\times$ 8) and the initial learning rate is (2.5e-5 $\times$ 8). For HRNet with input size 896, the batch size is (12 $\times$ 8) and the initial learning rate is (2.5e-5 $\times$ 8).

4.3. Experiment results

4.3.1 Single model results

We train different models in different input size settings and report results on the test set in Table 1.

method	input size	gaze error
iTracker	224	4.02
iTracker with dilated CNN	224	3.80
iTracker_MHSA	224	3.54
iTracker_MHSA	448	3.42
BoTNet	224	3.84
HRNet	224	3.84
HRNet	448	3.50
HRNet	640	**3.22**
HRNet	768	3.37
HRNet	896	3.39
ResNeSt	448	3.34

Table 1. Averaged angular errors of single models

method	input size	weight	gaze error
iTracker_MHSA	448	0.33	3.42
HRNet	640	0.33	3.22
ResNeSt	448	0.33	3.34
average of above	/	/	**3.14**

Table 2. Average angular errors of the ensemble results of 3 models

method	input size	weight	gaze error
iTracker_MHSA	448	0.2	3.42
BoTNet	224	0.1	3.84
HRNet	640	0.4	3.22
HRNet	768	0.1	3.37
HRNet	896	0.1	3.39
ResneSt	448	0.1	3.34
weighted average	/	/	**3.11**

Table 3. Average angular errors of the ensemble results of 6 models.

4.3.2 Ensembled model results

Experiments prove the weighted average of different single-model results is better than the best single model result in Table 1.Our ensembled model results is shown in Table 2 and Table 3. The weight of different models is chosen empirically and the lowest gaze error of our final result is 3.11.

References

[1] Jixu Chen and Qiang Ji. 3d gaze estimation with a single camera without ir illumination. In *2008 19th International Conference on Pattern Recognition*, pages 1–4. IEEE, 2008. 1

[2] Zhaokang Chen and Bertram E Shi. Appearance-based gaze estimation using dilated-convolutions. In *Asian Conference on Computer Vision*, pages 309–324. Springer, 2018. 1

[3] Yihua Cheng, Shiyao Huang, Fei Wang, Chen Qian, and Feng Lu. A coarse-to-fine adaptive network for appearance-based gaze estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10623–10630, 2020. 1

[4] Yihua Cheng, Feng Lu, and Xucong Zhang. Appearance-based gaze estimation via evaluation-guided asymmetric regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 100–115, 2018. 1

[5] Yihua Cheng, Xucong Zhang, Feng Lu, and Yoichi Sato. Gaze estimation by exploring two-eye asymmetry. *IEEE Transactions on Image Processing*, 29:5259–5272, 2020. 1, 2

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 4

[7] Sidney D'Mello, Andrew Olney, Claire Williams, and Patrick Hays. Gaze tutor: A gaze-reactive intelligent tutoring system. *International Journal of human-computer studies*, 70(5):377–398, 2012. 1

[8] Elias Daniel Guestrin and Moshe Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on biomedical engineering*, 53(6):1124–1133, 2006. 1

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[11] Robert JK Jacob and Keith S Karn. Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. In *The mind's eye*, pages 573–605. Elsevier, 2003. 1

[12] Ming Jiang and Qi Zhao. Learning visual attention to identify people with autism spectrum disorder. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3267–3276, 2017. 1

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[14] K. Krafka, A. Khosla, P. Kellnhofer, H. Kannan, S. Bhandarkar, W. Matusik, and A. Torralba. Eye tracking for everyone. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2176–2184, 2016. 1, 2

[15] Päivi Majaranta and Andreas Bulling. Eye tracking and eye-based human–computer interaction. In *Advances in physiological computing*, pages 39–65. Springer, 2014. 1

[16] Carlos H Morimoto and Marcio RM Mimica. Eye gaze tracking techniques for interactive applications. *Computer vision and image understanding*, 98(1):4–24, 2005. 1

[17] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769, 2016. 2

[18] Aravind Srinivas, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. Bottleneck transformers for visual recognition. *arXiv preprint arXiv:2101.11605*, 2021. 2, 3

[19] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. *arXiv preprint arXiv:1904.04514*, 2019. 2

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 2

[21] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 2

[22] Shuo Wang, Ming Jiang, Xavier Morin Duchesne, Elizabeth A Laugeson, Daniel P Kennedy, Ralph Adolphs, and Qi Zhao. Atypical visual saliency in autism spectrum disorder quantified through model-based eye tracking. *Neuron*, 88(3):604–616, 2015. 1

[23] Hirotake Yamazoe, Akira Utsumi, Tomoko Yonezawa, and Shinji Abe. Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions. In *Proceedings of the 2008 symposium on Eye tracking research & applications*, pages 245–250, 2008. 1

[24] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020. 2

[25] Xucong Zhang, Seonwook Park, Thabo Beeler, Derek Bradley, Siyu Tang, and Otmar Hilliges. Eth-xgaze: A large scale dataset for gaze estimation under extreme head pose and gaze variation. In *European Conference on Computer Vision*, pages 365–381. Springer, 2020. 1

[26] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Revisiting data normalization for appearance-based gaze estimation. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*, pages 1–9, 2018. 3

[27] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *IEEE transactions on pattern analysis and machine intelligence*, 41(1):162–175, 2017. 1