Learning-based Region Selection for End-to-End Gaze Estimation

Xucong Zhang[1]
xucong.zhang@inf.ethz.ch
Yusuke Sugano[2]
sugano@iis.u-tokyo.ac.jp
Andreas Bulling[3]
andreas.bulling@vis.uni-stuttgart.de
Otmar Hilliges[1]
otmar.hilliges@inf.ethz.ch

[1] Department of Computer Science, ETH Zurich, Zürich, Switzerland
[2] Institute of Industrial Science, The University of Tokyo, Tokyo, Japan
[3] Institute for Visualisation and Interactive Systems, University of Stuttgart, Stuttgart, Germany

Abstract

Traditionally, appearance-based gaze estimation methods use statically defined face regions as input to the gaze estimator, such as eye patches, and therefore suffer from difficult lighting conditions and extreme head poses for which these regions are often not the most informative with respect to the gaze estimation task. We posit that facial regions should be selected dynamically based on the image content and propose a novel gaze estimation method that combines the task of region proposal and gaze estimation into a single end-to-end trainable framework. We introduce a novel loss that allows for unsupervised training of a region proposal network alongside the (supervised) training of the final gaze estimator. We show that our method can learn meaningful region selection strategies and outperforms fixed region approaches. We further show that our method performs particularly well for challenging cases, i.e., those with difficult lighting conditions such as directional lights, extreme head angles, or self-occlusion. Finally, we show that the proposed method achieves better results than the current state-of-the-art method in within and cross-dataset evaluations.

1 Introduction

Appearance-based gaze estimation methods based on convolutional neural networks (CNNs) have recently surpassed classical methods, particularly for in-the-wild settings [31]. However, they are still not suitable for high-accuracy applications. Current CNN-based methods typically take either a single eye patch [6, 14, 25, 30, 34] or the eye region containing both eyes as input [4, 18]. While these approaches are sufficient for cases in which the face is mostly frontal and well lit, the question of which part of the image carries most of the information becomes important in uncontrolled settings with difficult lighting conditions and extreme head pose angles [19, 32]. For example, attempting to crop the two-eyes from a side-view or from an unevenly lit face is difficult at best and may result in non-informative

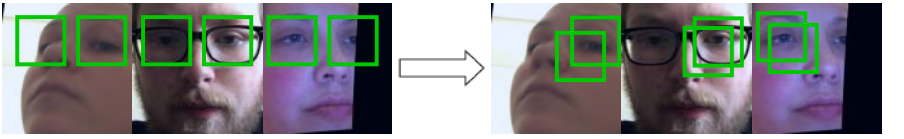

Figure 1: We propose a novel gaze estimation method that dynamically selects regions according to the properties of the input face image. Compared to selecting fixed eye regions (left), our method can select regions which are more informative, adapting to changing visibility and lighting conditions (right).

inputs (see Figure 1 for examples). Recent work has therefore proposed to leverage combinations of two eye patches in a probabilistic fashion [3] or to use the full face either alone [32] or in combination with eye patches [11].

We posit that the most informative regions in an image that is fed to a gaze estimator should be selected dynamically based on the image content. To this end we propose a novel gaze estimation method that combines the tasks of region selection and gaze estimation into a single end-to-end trainable framework. To the best of our knowledge, this is the first method that takes a dynamic region selection approach in the appearance-based gaze estimation task. The key technical challenge in learning to select good regions for gaze estimation is the mutual dependency between region selection and gaze estimation. For example, training a network to select good regions can be guided by the final gaze estimation accuracy. However, CNNs are usually sensitive to the type of images in the training data and typically do not generalize well to out-of-distribution samples. Therefore, once a gaze estimator has been trained with some hand-picked regions, a "better" crop may actually lead to reduced gaze estimation accuracy.

To address this issue we propose a training procedure in which a **R**egion **S**election **N**etwork (*RSN*) and the final gaze estimation network (*gaze net*) are trained in an alternating fashion. First, we train the *gaze net* by feeding randomly selected regions from a pool of potential region locations. Thus, the network learns to correlate input samples with gaze labels without over-fitting to the particular type of regions cropped from the input image. We then use this partially trained *gaze net* to train the *RSN* to select single or multiple regions from the source image. This process is guided by a novel loss that aligns the probability of picking a particular location with the gaze estimation error out of the *gaze net*. Once the *RSN* is fully trained, we re-train the *gaze net* to learn to predict more accurate gaze estimates given optimized region selections.

In summary, in this paper we contribute:

- A novel network architecture that combines a region selection and a gaze estimation network to dynamically select informative regions for gaze estimation.
- A three-stage training strategy alongside a novel loss to guide the training of the *RSN* module without the label.
- Experimental evidence that this approach leads to significant improvements compared to our own baseline as well as state-of-the-art static approaches on within GazeCapture and cross-dataset evaluations, particularly for challenging cases, e.g. difficult lighting conditions, extreme head angles, self-occlusion.

2 Related Work

Traditional approaches for remote gaze estimation often require specialised devices, use hand-crafted features, or perform model fitting [2, 24]. In contrast, appearance-based methods learn mappings between holistic eye appearances and gaze labels [6]. Appearance-based gaze estimation methods can take a variety of image patches as input. Most commonly a single eye-patch input [14, 16, 17, 21, 22, 25, 26, 27, 28, 30, 34, 35], which allows for the estimation of gaze from the left and right eyes of a person separately. Performance generally improves when considering both eye regions simultaneously [3, 4, 8, 18] or use multiple input regions, such as the two eyes alongside the face [4, 11]. However, it is unclear how to best hand-craft these specific regions, and how many regions should be selected. As shown in [3], some hand-picked regions may at times be unsuitable due to situational issues in illumination condition, image quality, and occlusions. Their suggested solution involves a separate evaluation network to select the better eye. Instead of hand-craft sub-regions, previous works also use the full-face input patches [9, 32] as single input. Such larger input regions may include factors which distract from the main task of gaze estimation. Recasens et al. [19] propose a model to learn the wrapping field that magnifies part of the input face image to perform gaze estimation. This method can aid the network in focusing on more important sub-regions. However, such magnification may destroy the geometry layout of the face which is critical for gaze estimation task. We propose a fully learning-based region selection framework for full-face appearance-based gaze estimation which only depends on the input face image. Our contribution eliminates the need to hand-craft the position of input regions, for gaze estimation networks.

Many attention mechanisms have been proposed for various computer vision tasks to attend to certain spatial regions. Recurrent neural networks are used to locate important regions for single object classification [15] with extensions to multiple objects [1], and dynamic decision on the RNN sequence length [13]. NTS-Net [29] employs a localization network ("Navigator"), and a classification network ("Scrutinizer") for the task of object classification where a "Teacher" network enforces ranking consistency between the Navigator-predicted informativeness of regions and the classification output of the Scrutinizer. In our setting, since gaze estimation is a regression task instead of classification, *gaze net* is sensitive to its input and cannot output the reasonable gaze estimation accuracy for the region it has not seen yet. Thus, *RSN* cannot use the *gaze net* net accuracy as a supervision signal. We solve this challenge with a three-stage training strategy, where the *gaze net* is first trained with random regions to act as a teacher network to the *RSN*.

3 Method

Most existing multi-region gaze estimation methods use image patches of the left and right eye as input [4, 11, 18]. We argue that in many cases the most informative sub-region of the face should be located dynamically based on head-pose, lighting and other extraneous factors (cf. Figure 1). As illustrated in Figure 2, the proposed method consists of two main components: *RSN* and *gaze net*. The *RSN* first takes a face image as input, from which it dynamically selects M regions based on a location pool. The goal of this sub-network is to identify the regions that are best suited for the task of gaze estimation. The *gaze net* then uses the selected regions together with a region grid that indicates the region's original location as features for gaze estimation. A subset of the general population exhibits non-agreeing

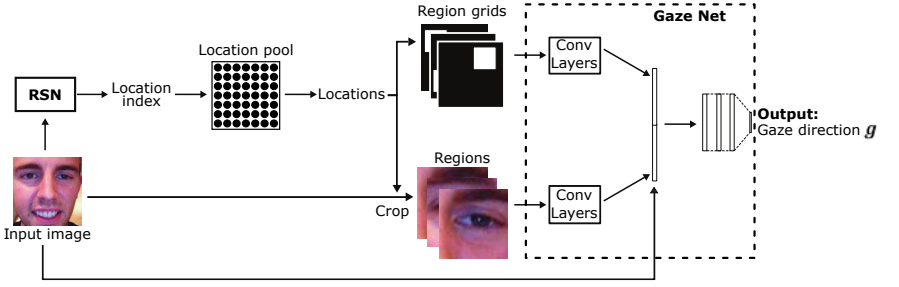

Figure 2: Method Overview. The *RSN* takes face image as input and outputs an index for the location pool. This location index is then mapped to a pixel location in the face image from which a surrounding region is cropped. The *gaze net* takes the cropped region image and the corresponding region grid as inputs. We extract features from the crop and the region grid and concatenate them. The combined feature vector is fed into three fully-connected layers to estimate a 2D gaze direction $\boldsymbol{g}$.

gaze directions from two eyes. Therefore, the *gaze net* can also take the full-face image as an additional input to improve performance in such cases.

One of the main difficulties in this approach is in designing an effective training procedure for the *RSN*. The reasons for this difficulty are two-fold. First, deciding which pixels are the most informative for gaze estimation task is a hard task even for a human annotator and hence there is no straightforward path to a fully-supervised scheme. Second, the *gaze net* accuracy cannot be directly used as a supervision signal to *RSN*, since the *RSN* will easily get stuck in local minima as it would first have to select regions that have higher gaze estimation errors than those on which the *gaze net* was originally trained. To alleviate these issues, we propose a strategy to first train the *gaze net* with randomly cropped image regions and use it as an imperfect but fair evaluator of candidate regions. In the following sections, we describe the details of our network architecture and the necessary training procedure.

3.1 Network architecture

Region Selection Network (RSN) We formulate the task of the *RSN* (denoted as a function, S) as a classification task, where the most appropriate location of one or more (M) fixed-size regions must be selected based on a pre-defined location pool. The location pool includes discrete locations denote the center of a potential region inside the input image. The number of candidate locations K in the location pool defines the search space of the *RSN*.

The *RSN* takes an input image $\boldsymbol{I}$ with 224×224 pixels as input and outputs a matrix of dimensionality $M \times K$ (in our experiments $M < 4$). Each element p_{mk} represents the probability of selecting the k-th location as the m-th region. The final location index for the m-th region is determined as $\hat{k}_m = \text{argmax}_{j \in \{1,\ldots K\}} p_{mj}$, which is used to crop the input image to yield $\boldsymbol{I}_{sub}$. The whole process can be written as $\boldsymbol{I}_{sub} = S(\boldsymbol{I})$. We compute the joint probability of all M regions as $p = \prod_{m=1}^{M} p_m$.

For training efficiency, we define the location pool to be a set of $7 \times 7 = 49$ uniformly distributed locations. We set the size of all regions to be roughly the size of one eye as 0.3 times the original face image size results in 68×68 pixels. The pre-defined location pool and the fixed region size constrain the search space of the *RSN* which drastically improves training convergence.

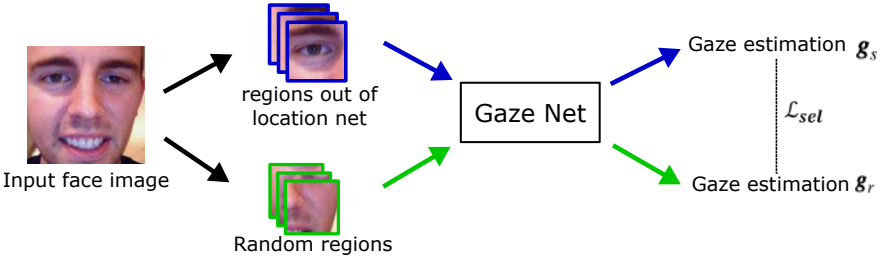

Figure 3: Training process for *RSN*. The *gaze net* takes the regions estimated by the *RSN* and outputs a gaze direction vector $\boldsymbol{g}_s$. The *gaze net* takes region according to a random selection from the location pool and outputs gaze direction vector $\boldsymbol{g}_r$. The gaze estimation error of $\boldsymbol{g}_s$ and $\boldsymbol{g}_r$ are then used to calculate the location loss $\mathcal{L}_{sel}$ to update *RSN*.

Gaze net The *gaze net* G accepts multiple regions $\boldsymbol{I}_{sub}$ and corresponding region grid as input and outputs a two-dimensional gaze direction $\boldsymbol{g}' = G(\boldsymbol{I}_{sub})$. All input region are resized to 224×224 pixels and are passed through the convolutional layers of one ResNet-18 networks. Note the channel numbers of the first convolutional layer is modified to accept multiple input images. To denote the region's original location in the input image we pass a region grid to the *gaze net*. The size of the region grid is 28×28 pixel with mostly zeros values except a 8×8 patch indicating the region's location. The region grid passes through two convolutional layers and two max-pooling layers which results in a $5 \times 5 \times 50 = 1,250$-dimensional feature vector. The features from each region are concatenated and then fed into one fully-connected layer to output the final gaze direction estimate $\boldsymbol{g} = (\phi, \theta)$. ϕ and θ represent yaw and pitch rotation angles in the spherical coordinate system.

After experimentally validating several CNN architectures including AlexNet [12] and InceptionNet [23], we use ResNet-18 [7] for both *RSN* and *gaze net* architectures with minimal modifications such as defining the final output dimensionality.

3.2 Training procedure

First, to train the network we propose a novel three-stage training strategy, the *gaze net* is trained by randomly selecting locations from the location pool to crop regions and evaluate the gaze estimation loss $\mathcal{L}_{gaze}$. Training with random crops prevents overfitting to a particular type of region. Assuming there are N training samples and we have ground-truth gaze direction vectors $\boldsymbol{g}_i$ for the i-th sample, the gaze estimation loss $\mathcal{L}_{gaze}$ is given by

$$\mathcal{L}_{gaze} = \frac{1}{N} \sum_{i=1}^{N} \left| \boldsymbol{g}_i - \boldsymbol{g}_i' \right|. \tag{1}$$

Since this is a fully-supervised procedure, it can be assumed that the attained gaze estimation accuracy is a good proxy for the utility of the region.

Second, we leverage this assumption to train the *RSN* based on the gaze estimates of the initial *gaze net*. As shown in Figure 3, for this purpose we evaluate two different sets of regions in the forward pass. We use regions $\boldsymbol{I}^s_{sub}$ proposed by the *RSN*, and another set of randomly selected regions $\boldsymbol{I}^r_{sub}$ at different locations. The *gaze net* outputs gaze direction vectors $\boldsymbol{g}_s$ and $\boldsymbol{g}_r$ for regions $\boldsymbol{I}^s_{sub}$ and $\boldsymbol{I}^r_{sub}$, respectively.

We denote the probability of selecting $\boldsymbol{I}^s_{sub}$ and $\boldsymbol{I}^r_{sub}$ estimated by the *RSN* as p_s and

	Single-region	Two-region	Three-region	Two-region + Face	Three-region + Face
Baseline (w/o *RSN*)	4.58°	3.75°	-	3.58°	-
Ours	**4.25°**	**3.60°**	**3.43°**	**3.51°**	**3.32°**

Table 1: Comparison of our model against a baseline model without *RSN*, in the within-GazeCapture evaluation setting. For the single region model, the baseline takes either left or right eye images as input to the *gaze net*. For the two-region model, the baseline takes two eye images as input to the *gaze net*. We also show results when supplying the face image as additional input. Ours consistently outperforms the baseline.

p_r. The goal is to link the gaze estimation errors $\boldsymbol{g}_s$ and $\boldsymbol{g}_r$ with the corresponding location probabilities. Aligning these quantities then provides a supervision signal for *RSN*.

The gaze estimation errors e_s and e_r for both $\boldsymbol{g}_s$ and $\boldsymbol{g}_r$ are then computed. To train the *RSN* with the precise physical error metrics, we use the angular gaze estimation error instead of the approximate L1 loss as $\mathcal{L}_{gaze}$. To calculate the angular gaze estimation error, we first convert the yaw and pitch angles $\boldsymbol{g} = (\phi, \theta)$ into three-dimensional representation in the Cartesian coordinate system as $\boldsymbol{v} = (\cos\phi\cos\theta, -\sin\phi, \cos\phi\sin\theta)$. The angular error e between ground-truth $\boldsymbol{v}$ and prediction $\hat{\boldsymbol{v}}$ is defined as $e = \arccos \frac{\boldsymbol{v}^\mathsf{T}\cdot\hat{\boldsymbol{v}}}{|\boldsymbol{v}|\cdot|\hat{\boldsymbol{v}}|}$ The training objective for *RSN* is to output selection probabilities for the two regions that correspond to the gaze error. We formulate the following loss to enforce that the probabilities become proportional to their corresponding gaze estimation errors

$$\mathcal{L}_{sel} = \frac{1}{N}\sum_{i=1}^{N}\left|\min\left(\frac{p_s}{p_r}, \delta\right) - \frac{e_r}{e_s}\right|_i, \tag{2}$$

where i indicate the indices for training samples. We experimentally found that the ratio p_s/p_r can be excessively large to cause large gradient updates and resulting in oscillations of $\mathcal{L}_{sel}$. To address this issue by constraining the gradient update, we use clipped surrogates [20] with threshold δ. We set the threshold $\delta = 3.0$ in our implementation.

After convergence of the *RSN*, we train the *gaze net* from scratch, now with regions suggested by the *RSN* (in some experimental settings we also provide the face image as $\boldsymbol{g} = G([S(\boldsymbol{I}), \boldsymbol{I}])$). While it is also possible to fine-tune the initial *gaze net*, we opt to train the *gaze net* from scratch to ensure fair comparison with baselines during experiments.

4 Experiments

In this section, we discuss our experiments conducted to assess the effectiveness of the proposed method. We first perform experiments with a single-region model, contrasting to a strong baseline and to evaluate the region selection strategy in Sec. 4.1. We then increase the number of regions proposed by *RSN* in Sec. 4.2. Finally, we compare our method with the current state-of-the-art in Sec. 4.3.

Datasets. We used GazeCapture [11], EYEDIAP [5] and MPIIGaze [34] datasets for the evaluation. We pre-processed these datasets following the data normalization procedure described in [33] to extract the face image and the corresponding gaze direction labels. In short, the data normalization procedure places a virtual camera to re-render the eye image

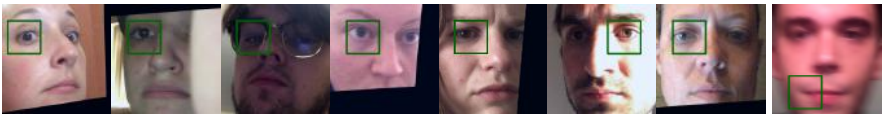

Figure 4: Examples of the selected regions for the single-region model. The green rectangle indicates the selected region. Our model selects the eye with better visibility (a-d) or lighting condition (e-g). Extreme motion blur is one failure case (h).

from a reference point with the head upright, which results in normalized face images without any in-plane rotation. Since the GazeCapture dataset only provides gaze labels on a 2D screen, we used the pre-processing pipeline from [18] to attain 3D head pose from GazeCapture. Note that the same ground-truth gaze vectors in the normalized face coordinate system are always used in the following experiments, including the single-eye baseline methods. We performed most of our experiments on the GazeCapture dataset which includes more data and appearance variations than EYEDIAP and MPIIGaze. We used both phone and tablet sessions in GazeCapture, and the pre-defined training and test split from [11]. We selected screen target (CS/DS) and static head pose sequences (S) from the EYEDIAP dataset, sampling every 15 seconds from its VGA video streams.

Training details. We used the Adam [10] optimizer with initial learning rate 1×10^{-4}, batch size 90, and with the momentum values set to $\beta_1 = 0.9$ and $\beta_2 = 0.95$. The *gaze net* was trained for 25 epochs with random regions, and then the *RSN* was trained for 25 epochs. Finally, the *gaze net* was re-trained for 25 epochs with the regions proposed by *RSN*. We stopped the training after 25 epochs, equating to two days of training with a modest GPU. The learning rate was decayed with a factor of 0.1 for every 10 epochs. The baselines with eye patches as regions were trained with the same network architecture as the *gaze net* for 20 epochs. For data augmentation, we applied random horizontal and vertical translations to the the input image with a factor of 0.3 times the image size.

4.1 Single-region selection

We experimented first with the most simple condition, where the *RSN* outputs only a single region. We designed our baselines to be exact re-implementations of the prior art. Previous works have either mirrored the right eye to the left, to double the training data [25, 26, 30, 34], or they have trained two separate models for the left and right eyes and averaged the independent predictions afterwards [6, 14]. We choose the latter approach for our single-region baseline. This choice keeps the number of training samples consistent across baselines, to ensure a fair comparison. Specifically, the baseline is formed by training two *gaze nets* with a left and right eye respectively, and then computing the average of the estimates. The eye region was cropped by data normalization with facial landmarks.

The results are summarized in the first column of Table 1. We can see that our method achieves better performance (4.25 degrees) than the baseline single-region network (4.58 degrees) with significant margin (7.2%). We show some qualitative examples of the selected regions in Figure 4. In general, the *RSN* selects either the left or right eye region since they are naturally the most informative regions. Among two eyes, the *RSN* tends to pick the eye with the better visibility by taking into account self-occlusion (Figure 4 (a,b)), reflections on the surface of eyeglasses (Figure 4 (c)), the facial region being out of the frame (Figure 4 (d,e)), or the brighter eye under directional lighting conditions (Figure 4 (f,g)). On the

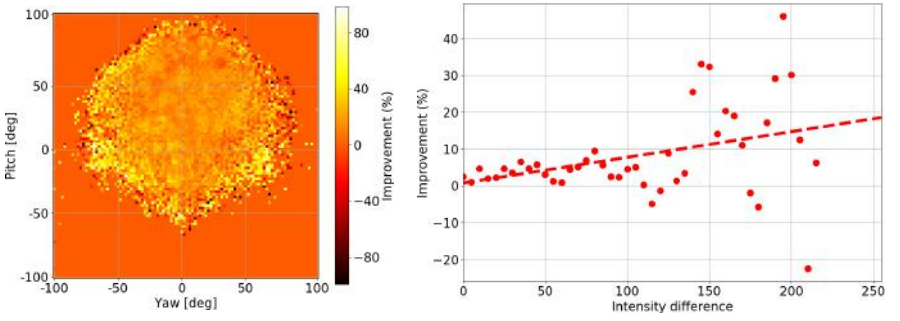

Figure 5: Left: improvement of our model over the baseline in percentage across the head pose distribution. The x-axis and y-axis are the yaw and pitch head rotations. Larger improvements can be observed under extreme head poses. Right: improvement of our model over the baseline across horizontal differences between left and right parts of the face. The dots are median improvements and the dashed line shows a linear polynomial curve fit. Larger improvements happen under more challenging large intensity differences.

other hand, the *RSN* can fail at selecting the eye regions in challenging cases such as when provided with low-quality blurry face images (Figure 4 (h)).

Robustness to hard cases In the left of Figure 5, we further show the performance gain by our model with respect to head poses. The x-axis and y-axis correspond to the yaw and pitch head rotations, and the heatmap represents the color-coded performance improvement from the baseline method in percentage. It can be seen that larger improvements are achieved under extreme head poses which typically cause self-occlusions. Similarly, we visualize the improvement with respect to different lighting conditions in the right of Figure 5. The horizontal axis corresponds to the difference between the mean intensity values of the left and right sides of the face image, and a larger difference indicates that there is a strong directional light. The dots are median improvements with a window of five (intensity differences), and the dashed line shows a linear polynomial curve fit. The performance improvement depicted in the figure is not stable since fewer data samples result in higher variance for large intensity differences. As we can see, our model achieves larger performance improvements on images with stronger directional lighting.

4.2 Multi-region selection

We now report experimental results when the *RSN* outputs multiple regions. For the baseline, we pass both left and right eye regions as input to the *gaze net*, which resembles the previous works using two eyes as input [4, 18]. In contrast, our model takes two or three regions provided by the *RSN* as input to the *gaze net*. Results are summarized in the second and third columns of Table 1. Our two-region model achieves better performance (3.60 degrees) than the baseline (3.75 degrees), which indicates the advantage of selecting regions dynamically instead of using fixed crops. Increasing the number of regions to three achieves the best results (3.43 degrees) with 8.5% significant improvement (paired t-test: $p < 0.01$) over the baseline (3.75 degrees), and demonstrates the potential of our approach.

Examples of the regions selected by the two-region and three-region *RSN* are shown in the first and second rows in Figure 6, respectively. In general, for most input images, our

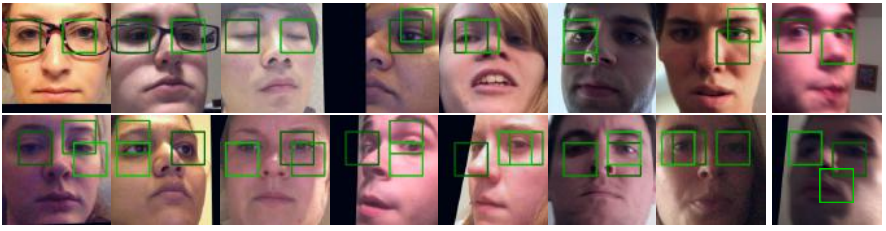

Figure 6: Examples of region selection for the two-region (first row) and three-region (second row) cases. The green rectangles indicate selected regions. Our model selects regions that contain both eyes for most of the samples (a-c), focuses on a single eye due to visibility (d,e) or brightness (f,g). Extreme motion blur can cause failure in selecting eyes (h).

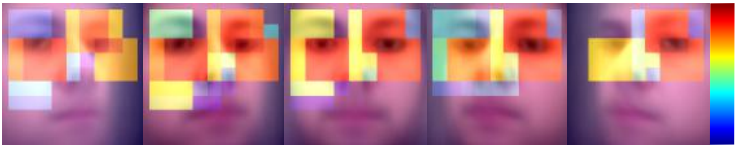

Figure 7: Spatial distributions of selected regions by our three-region model for different horizontal head angles. We visualize the regions' locations using the jet color map and overlay the heat maps on the averaged face images corresponding to each pose.

model picks regions around the left and right eye, sometimes with a small offset (Figure 6 (a-c)). Even with the multi-region selection strategy our model sometimes focuses on a single eye due to visibility (Figure 6 (d,e)) or brightness (Figure 6 (f,g)). Under extreme blur, the *RSN* selects regions away from the eye regions (Figure 6 (h)). We further show the spatial distribution of the selected regions over different horizontal head angles by our three-region model in Figure 7. We overlay the region-distribution heat maps over the averaged face images. From these examples, we can see that our model can select the eye with better visibility without self-occlusion. Such a trend becomes more evident for extreme head poses.

We examined the effect of providing the face image as additional input (*e.g.*, [11]) to our model by treating the face image as one of the inputs for the *gaze net*. Note that only taking the single face-patch as input results in 4.21 degrees error, which is worse than two-region models. The results are summarized in the last two columns of Table 1. By adding the face image input, the baseline model benefits more (4.5% from 3.75 degrees to 3.58 degrees) than our two-region model (2.6% from 3.40 degrees to 3.31 degrees). One of the possible reasons is that the selected regions by the *RSN* already cover the necessary information for the task. Therefore, the additional pixels do not improve estimates for our method as much as in the case of the baseline, where the crop is fixed and hence informative regions may be omitted. Our three-region model still achieves the best overall performance with 3.32 degrees, which again suggests that optimized region selection is helpful.

4.3 Comparison with the state of the art

In our final experiment, we compare to the current state-of-the-art methods [9, 16, 17, 18, 32] in within MPIIGaze (15-fold), within EYEDIAP (5-fold), within GazeCapture, cross-dataset from GazeCapture to MPIIIGaze, and cross-dataset from GazeCapture to EYEDIAP evalua-

	EYEDIAP	MPIIGaze	GazeCapture	GazeCapture → MPIIGaze	GazeCapture → EYEDIAP
[32]	**6.0°**	4.8°	-	-	-
[4]	-	4.8°	-	-	-
[16]	10.3°	**4.5°**	-	-	-
[18]	-	-	3.5°	5.2°	-
[17]	7.1°	-	-	-	-
Baseline	7.3°	5.0°	3.6°	5.4°	6.3
Ours	6.6°	**4.5°**	**3.3°**	**4.9°**	**6.0**

Table 2: Comparison with the state of the art. Ours consistently outperforms the baseline model and is equal (MPIIGaze) or better than the state-of-the-art, except for the EYEDIAP dataset. Note that both MPIIGaze and EYEDIAP are small datasets and don't contain sufficient samples to learn both region selection and gaze estimation. Note that ours performs particularly well in the most challenging cross-dataset setting when trained on the large GazeCapture dataset even if tested on MPIIGaze or EYEDIAP.

tions. We used the result from our Three-region + Face model in our experiments. Although [18] mostly focuses on the few-shot person-specific task, it sets the current state-of-the-art for within GazeCapture and cross-dataset from GazeCapture to MPIIGaze evaluations. Our comparison with [18] is a fair comparison since the experimental setting without calibration in Fig. 6 of [18] is the same as ours for the person-independent case. For the within EYEDIAP and within MPIIGaze evaluation, our method consistently outperforms the baseline and achieves comparable results with respect to previous state-of-the-art methods. We note that our method requires training of two networks and hence requires large training datasets. Neither EYEDIAP nor MPIIGaze are particularly large which prevents our model from significantly outperforming the SoA in this setting. However, when leveraging large amounts of data such as GazeCapture our method outperforms [18] and the baseline (w/o RSN) on the more challenging cross-dataset settings. A more sample efficient approach to the dynamic selection module is another important direction for future work.

5 Conclusion

In this paper, we ask the questions: which part of an image is most informative towards the task of eye gaze estimation, and how can we automatically extract these regions from the input image? To address these questions we propose a novel architecture to jointly learns the tasks of region selection and gaze estimation. Our core technical contribution is a novel loss that aligns the probabilities of selecting a particular region with the final gaze estimation loss. We empirically show that i) our technique learns meaningful region selection strategies, such as picking the better illuminated eye, ii) dynamic region selection leads to better gaze estimation performance compared to static heuristics, and iii) this improvement is particularly pronounced for difficult cases, including extreme head-angles and self-occlusions.

6 Acknowledgements

A. Bulling was supported by the European Research Council (ERC; grant agreement 801708).

References

[1] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

[2] Jixu Chen and Qiang Ji. Probabilistic gaze estimation without active personal calibration. In *CVPR 2011*, pages 609–616. IEEE, 2011.

[3] 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.

[4] Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. Rt-gene: Real-time eye gaze estimation in natural environments. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 334–352, 2018.

[5] Kenneth Alberto Funes Mora, Florent Monay, and Jean-Marc Odobez. Eyediap: A database for the development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, pages 255–258. ACM, 2014.

[6] Dan Witzner Hansen and Qiang Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32 (3):478–500, 2010.

[7] 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.

[8] Qiong Huang, Ashok Veeraraghavan, and Ashutosh Sabharwal. Tabletgaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. *Machine Vision and Applications*, 28(5-6):445–461, 2017.

[9] Petr Kellnhofer, Adria Recasens, Simon Stent, Wojciech Matusik, and Antonio Torralba. Gaze360: Physically unconstrained gaze estimation in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6912–6921, 2019.

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

[11] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye tracking for everyone. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2176–2184, 2016.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[13] Zhichao Li, Yi Yang, Xiao Liu, Feng Zhou, Shilei Wen, and Wei Xu. Dynamic computational time for visual attention. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1199–1209, 2017.

[14] Gang Liu, Yu Yu, Kenneth Alberto Funes Mora, and Jean-Marc Odobez. A differential approach for gaze estimation with calibration. In *BMVC*, volume 2, page 6, 2018.

[15] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pages 2204–2212, 2014.

[16] Seonwook Park, Adrian Spurr, and Otmar Hilliges. Deep pictorial gaze estimation. In *European Conference on Computer Vision (ECCV)*, ECCV '18, 2018.

[17] Seonwook Park, Xucong Zhang, Andreas Bulling, and Otmar Hilliges. Learning to find eye region landmarks for remote gaze estimation in unconstrained settings. In *ACM Symposium on Eye Tracking Research and Applications (ETRA)*, ETRA '18, New York, NY, USA, 2018. ACM.

[18] Seonwook Park, Shalini De Mello, Pavlo Molchanov, Umar Iqbal, Otmar Hilliges, and Jan Kautz. Few-shot adaptive gaze estimation. In *International Conference on Computer Vision (ICCV)*, 2019.

[19] Adria Recasens, Petr Kellnhofer, Simon Stent, Wojciech Matusik, and Antonio Torralba. Learning to zoom: a saliency-based sampling layer for neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 51–66, 2018.

[20] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[21] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2107–2116, 2017.

[22] Yusuke Sugano, Yasuyuki Matsushita, and Yoichi Sato. Learning-by-synthesis for appearance-based 3d gaze estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1821–1828, 2014.

[23] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[24] Kang Wang and Qiang Ji. Real time eye gaze tracking with 3d deformable eye-face model. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1003–1011, 2017.

[25] Kang Wang, Rui Zhao, and Qiang Ji. A hierarchical generative model for eye image synthesis and eye gaze estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 440–448, 2018.

[26] Kang Wang, Rui Zhao, Hui Su, and Qiang Ji. Generalizing eye tracking with bayesian adversarial learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11907–11916, 2019.

[27] Erroll Wood, Tadas Baltrusaitis, Xucong Zhang, Yusuke Sugano, Peter Robinson, and Andreas Bulling. Rendering of eyes for eye-shape registration and gaze estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3756–3764, 2015.

[28] Yunyang Xiong, Hyunwoo J Kim, and Vikas Singh. Mixed effects neural networks (menets) with applications to gaze estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7743–7752, 2019.

[29] Ze Yang, Tiange Luo, Dong Wang, Zhiqiang Hu, Jun Gao, and Liwei Wang. Learning to navigate for fine-grained classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 420–435, 2018.

[30] Yu Yu, Gang Liu, and Jean-Marc Odobez. Improving few-shot user-specific gaze adaptation via gaze redirection synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019.

[31] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4511–4520, 2015. doi: 10.1109/CVPR.2015.7299081.

[32] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It's written all over your face: Full-face appearance-based gaze estimation. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 2299–2308. IEEE, 2017.

[33] 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*, page 12. ACM, 2018.

[34] 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, 2019.

[35] Wangjiang Zhu and Haoping Deng. Monocular free-head 3d gaze tracking with deep learning and geometry constraints. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3143–3152, 2017.