Supplementary Material

1 Data Curation Cost

The data collection process requires either manual or sensor based annotations (See Table 1). As this being time consuming process, the research community moves towards the data generation process for benchmarking with a large variation in data attributes. Prior works in this domain generate both synthetic and real image. In order to capture the possible rotational variation in image, gaze redirection techniques [4,5,13,14] are quite popular. The other approaches are mainly based on random forest [8] and style transfer [11]. However, due to several image quality based limitations, these generated datasets are not used for benchmarking.

2 Experimental Details

2.1 Automatic 3-frame set mining on Benchmark Datasets

For generating 3-frame sets, following process is followed:
CAVE [12] dataset is collected with 7 horizontal and 3 vertical gaze locations as shown in the left part of Fig. 1. Considering these positions as 7 × 3 grids, we defined three types of gaze trajectories: horizontal, vertical and diagonal. As temporal information is missing, we can consider bi-directional gaze trajectories. We reverse the order for bidirectional set mining (Refer Fig. 1). The bidirectional gaze trajectories are applied for CAVE dataset only due to the absence of temporal information. Note that this does not impact the requirement of ground truth annotation for weak supervision. In this way, we collect 3,024 3-frame sets for training. For this dataset, we require 6.56% and 3.28% of prior data annotation for our '2-labels' and '1-label' paradigms.
TabletGaze [6] dataset is also collected in a 7 × 5 grid format (as shown in the middle image of Fig. 1). Similar to CAVE dataset, we define horizontal, vertical and diagonal gaze trajectories and collect 108,524 3-frame sets. For TabletGaze dataset, as temporal information is also present, we consider unidirectional frames only. For 3-frame set mining on TabletGaze data, we require less than 1% prior data annotation for both the frameworks.

Table 1: Comparison of benchmark datasets for cost analysis.

Dataset	Cost Analysis
CAVE [12]	Canon EOS Rebel T3i camera and a Canon EF-S 18–135 mm IS f/3.5–5.6 zoom lens
MPII [17]	Laptop Collection duration: 3 months
TabletGaze [6]	Samsung Galaxy Tab S
Gaze360 [7]	Ladybug5 360°panoramic camera, AprilTag
ETH-XGaze [16]	18 Canon 250D SLR camera, ESPER trigger box, Raspberry Pi and with controlled illumination.

MPII [15] gaze dataset is collected by showing random points on the laptop screen to the participants. To make the gaze trajectory smooth, we sort the given coordinates of the points in ascending order and consider it as a gaze trajectory. Further, 3-frame sets are collected in a day-wise for each participant. Following this procedure, we collect 32,751 3-frame sets. We extracted these 3-frame sets with 4.67% prior data annotation.
For **Gaze360** [7], we compute person-specific 3-frame sets. As each participant fixates gaze at a moving target, we consider the target's trajectory as the gaze trajectory. This results in 197,588 3-frame sets and we use 2.38% of annotated data.

2.2 On unlabelled 'in the wild' YouTube data

We evaluate our method on an 'in the wild' data i.e. when the expert/ground truth labels are not available. We collect approximately 400,000 frames from YouTube videos.

Gaze Trajectory Selection and 3-frame set Mining As the relative positions of pupil-centers provide the most important information regarding gaze direction, we utilize this property to detect gaze trajectories. We utilize two eye symmetry property i.e. the change in relative position of the iris is symmetrical while scanning 3D space [2]. Based on this hypothesis, we compare the vertical angles formed with the following points i.e. pupil-center, nose in both eyes. In Fig. 3, I_1 and I_2 are the pupil centers; V is the vertical direction w.r.t. the nose tip point and θ_1, θ_2 are the above mentioned angles. The change in θ_1 and θ_2 depicts the path of the gaze trajectory sequence. For example, if a person shifts his/her gaze from left to right, the values of the angles will be as follows: initially, θ_1 will be greater than θ_2; then gradually θ_1 will decrease and θ_2 will increase; finally, θ_2 will be greater than θ_1. Thus, by monitoring these angles we can approximate gaze trajectories. The heuristic also considers the trajectory segment if it starts from the middle until there is a change in any of the angles θ_1 and θ_2. Although the proposed method is robust to head movements within the range of $-10°$ to $10°$. After identifying the gaze trajectories, we annotate the start and end frames with OpenFace [1] and collect possible 3-frame sets. In this 3-frame set collection and data annotation procedure, we require 5.34% and

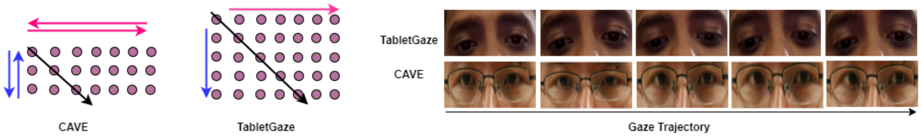

Fig. 1: 3-frame set mining process for CAVE [12] and TabletGaze dataset [6]. Here, red, blue and black arrows represent horizontal, vertical and diagonal gaze trajectories.

2.67% annotation of the overall data for '2-labels' and '1-label' settings. According to literature [3], human gaze trajectories are considered to be spherical. Thus for ground truth annotation, we label remaining frames by SLERP interpolation method [3]. Further, we apply our '2-label' and '1-label' model on this data.

2.3 Evaluation Metrics

For quantitative evaluation, we use Mean Absolute Error (MAE), Correlation Coefficient (CC) and Angular Error. Mean Absolute Error is calculated as: $\frac{\sum_{i=1}^{n}|y^p - y|}{n}$ Correlation coefficient is calculated as: $\frac{\sum_{i=1}^{n}(y_i-\overline{y})(y_i^p-\overline{y^p})}{\sqrt{\sum_{i=1}^{n}(y_i-\overline{y})^2 \ \sum_{i=1}^{n}(y_i^p-\overline{y^p})^2}}$ Here, y^p is the predicted label and y is the ground truth label in normalized space, $(\overline{\cdot})$ indicates mean across the samples. Similar to the previous methods [16,10,9], angular error is the average error across test data measured in terms of cosine angle between ground truth and predicted gaze direction. It is measured as follows: $\frac{g}{||g||_2} \cdot \frac{g'}{||g'||_2}$ Here, g and g' respectively denote the ground truth and predicted gaze in terms of 3D gaze direction vector.

Table 2: Cross dataset performance evaluation among the benchmark datasets *in terms of MAE* in both '2-labels' and '1-label' settings. Given any two datasets D_1 and D_2, the training is performed on the train partition of D_1 (i.e. D_1^{train}) using the Original Label (OL) and ResNet-50 as backbone network. Further, it is evaluated on test partition of D_2 (i.e. D_2^{test}).

Original Label	**2-labels** **Test→** **Train** **↓**	**CAVE**	**MPII**	**Gaze360**	**TabletGaze**
	CAVE	–	0.50	0.55	0.49
	MPII	0.35	–	0.53	0.51
	Gaze360	0.21	0.40	–	0.44
	TabletGaze	0.36	0.42	0.50	–

Original Label	**1-label** **Test→** **Train** **↓**	**CAVE**	**MPII**	**Gaze360**	**TabletGaze**
	CAVE	–	0.54	0.60	0.50
	MPII	0.57	–	0.61	0.90
	Gaze360	0.24	0.39	–	0.50
	TabletGaze	0.27	0.49	0.59	–

Table 3: Cross dataset performance evaluation among the benchmark datasets and YouTube data *in terms of MAE* in both '2-labels' and '1-label' settings. Given any two datasets D_1 and D_2, the training is performed on the train partition of D_1 (i.e. D_1^{train}) using the Predicted Label (PL) i.e. output of '2-labels' technique Y_{ul}^{p}. Further, it is evaluated on the test partition of D_2 (i.e. D_2^{test}).

Predicted Label	**2-labels Test→ Train ↓**	**CAVE**	**MPII**	**Gaze360**	**TabletGaze**
	CAVE	–	0.54	0.51	0.47
	MPII	0.27	–	0.50	0.48
	Gaze360	0.20	0.34	–	0.42
	TabletGaze	0.32	0.44	0.50	–
	YouTube	0.25	0.45	0.46	0.43

Predicted Label	**1-label Test→ Train ↓**	**CAVE**	**MPII**	**Gaze360**	**TabletGaze**
	CAVE	–	0.55	0.54	0.50
	MPII	0.30	–	0.52	0.50
	Gaze360	0.23	0.37	–	0.49
	TabletGaze	0.25	0.46	0.53	–
	YouTube	0.30	0.52	0.49	0.45

3 Results

3.1 Cross Dataset Evaluation

We perform a cross dataset evaluation for predicting the generalization ability of the proposed method. This evaluation is conducted in two settings: The first configuration is a classical cross dataset evaluation protocol. In the second configuration, the training is performed on the train partition of OD using the PL i.e. output of '2-labels' technique Y_{ul}^{p} and it is evaluated on the test partition of other datasets. The second configuration is conducted for cross dataset label quality assessment of the proposed method.

The first configuration is a classical cross dataset evaluation. Given any two datasets D_1 and D_2, the training is performed on the train partition of D_1 (i.e. D_1^{train}) using the OL and ResNet-50 as backbone network. Further, it is evaluated on test partition of D_2 (i.e. D_2^{test}). The results are shown in Table 2. In the second configuration, the training is performed on the train partition of D_1 (i.e. D_1^{train}) using the PL i.e. output of '2-labels' technique Y_{ul}^{p}, while it is evaluated on the test partition of D_2 (i.e. D_2^{test}). The results are depicted on Table 3. Please note that CAVE, TabletGaze and MPII do not have proper train-validation-test partitions [12,15,6]. We train the model on D_1 and evaluate on D_2. From both the Tables 2 and 3, it is observed that with the predicted label,

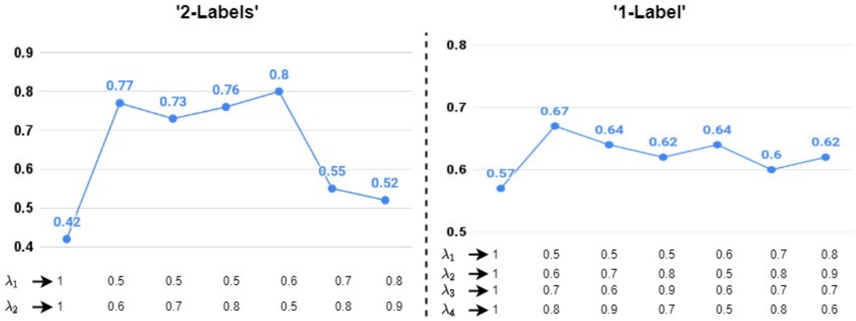

Fig. 2: Impact of regularization parameters in ‘2-labels’ and ‘1-label’ settings.

the cross dataset performance is increased significantly which in turn, shows the generalization capability of our models. Apart from generalizability, it is also observed that the ‘1-label’ framework performs comparatively well even after it is trained with less supervision.

For ‘in-the-wild’ data, the cross dataset performance is depicted in Table 3 for both 2-labels and 1-label frameworks. Here (Table 3), we have not fine-tuned the model.

3.2 Ablation Studies

‘Resnet-50+FC’ Vs ‘2-labels’. We also evaluate our method against a simple ‘ResNet+FC’ trained on training partition and tested on the test partition. The MAE is 0.39 as compared to our ResNet-50 based ‘2-labels’ framework (i.e. 0.25). This experiment also indicates the advantage of using triplet module.

Regularization Parameters. We have also experimented with different values of the regularization parameters. The trade-off is shown in Fig. 2. In this figure, the overall loss is plotted against different values of λ. It indicates that the optimal setting is achieved when all values are 1.

Computational Complexity. Quantitatively, for real world data, it takes 15 seconds (on an average) for all tasks including inference.

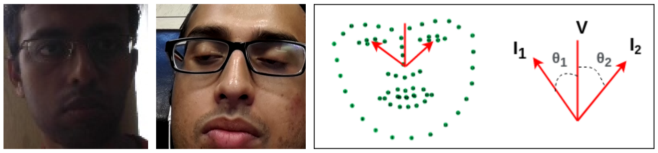

Fig. 3: (Left) Two sample images for which our methods generate noisy labels due to illumination conditions and eye openness. (Right) Heuristic for gaze trajectory selection.

3.3 Failure Cases

We also investigate the failure cases of our methods. The generated labels are noisy when the illumination is dark and the eyes are not open. Fig. 3 shows a few cases, where the correlation is low as compared to the ground truth labels.

References

1. Baltrušaitis, T., Robinson, P., Morency, L.P.: Openface: an open source facial behavior analysis toolkit. In: IEEE Winter Conference on Applications of Computer Vision. pp. 1–10 (2016)
2. Dubey, N., Ghosh, S., Dhall, A.: Unsupervised learning of eye gaze representation from the web. In: 2019 International Joint Conference on Neural Networks (IJCNN). pp. 1–7. IEEE (2019)
3. Eberly, D.: A fast and accurate algorithm for computing slerp. Journal of Graphics, GPU, and Game Tools **15**(3), 161–176 (2011)
4. Ganin, Y., Kononenko, D., Sungatullina, D., Lempitsky, V.: Deepwarp: Photorealistic image resynthesis for gaze manipulation. In: European conference on computer vision. pp. 311–326. Springer (2016)
5. He, Z., Spurr, A., Zhang, X., Hilliges, O.: Photo-realistic monocular gaze redirection using generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6932–6941 (2019)
6. Huang, Q., Veeraraghavan, A., Sabharwal, A.: Tabletgaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. Machine Vision and Applications **28**(5-6), 445–461 (2017)
7. Kellnhofer, P., Recasens, A., Stent, S., Matusik, W., Torralba, A.: Gaze360: Physically unconstrained gaze estimation in the wild. In: IEEE International Conference on Computer Vision (2019)
8. Kononenko, D., Lempitsky, V.: Learning to look up: Realtime monocular gaze correction using machine learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4667–4675 (2015)
9. Park, S., Mello, S.D., Molchanov, P., Iqbal, U., Hilliges, O., Kautz, J.: Few-shot adaptive gaze estimation. In: IEEE International Conference on Computer Vision. pp. 9368–9377 (2019)
10. Park, S., Zhang, X., Bulling, A., Hilliges, O.: Learning to find eye region landmarks for remote gaze estimation in unconstrained settings. In: Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications. pp. 1–10 (2018)
11. Sela, M., Xu, P., He, J., Navalpakkam, V., Lagun, D.: Gazegan-unpaired adversarial image generation for gaze estimation. arXiv preprint arXiv:1711.09767 (2017)
12. Smith, B., Yin, Q., Feiner, S., Nayar, S.: Gaze locking: passive eye contact detection for human-object interaction. In: ACM User Interface Software & Technology (2013)
13. Wood, E., Baltrušaitis, T., Morency, L.P., Robinson, P., Bulling, A.: Gazedirector: Fully articulated eye gaze redirection in video. In: Computer Graphics Forum. vol. 37, pp. 217–225. Wiley Online Library (2018)
14. Yu, Y., Liu, G., Odobez, J.: Improving few-shot user-specific gaze adaptation via gaze redirection synthesis. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 11937–11946 (2019)

15. Zhang, X., Sugano, Y., Fritz, M., Bulling, A.: Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)
16. Zhang, X., Park, S., Beeler, T., Bradley, D., Tang, S., Hilliges, O.: Eth-xgaze: A large scale dataset for gaze estimation under extreme head pose and gaze variation. In: European Conference on Computer Vision. pp. 365–381. Springer (2020)
17. Zhang, X., Sugano, Y., Fritz, M., Bulling, A.: Appearance-based gaze estimation in the wild. In: IEEE Computer Vision and Pattern Recognition. pp. 4511–4520 (2015)