

SG-GAZE: STRUCTURALLY AND GEOMETRICALLY CONSISTENT REPRESENTATION LEARNING FOR GENERALIZABLE 3D GAZE ESTIMATION

Anonymous authors

Paper under double-blind review

A APPENDIX

Section B introduces our demo video information. A live demo of our method can be found at this link. Section C describes the implementation process of the method in more details. We have carried out additional experiments, and the results will be explained in Section D. Finally, in Section E, we will discuss the limitations of the method and future work. Our code will be released on GitHub upon acceptance.

B DEMO VIDEO

To better illustrate our proposed method, we provide a demo video as supplementary material. The video demonstrates the complete pipeline of our 3D gaze estimation framework, including raw eye image input, 2D rendering, 3D eyeball reconstruction, and mesh visualization. Specifically:

- **Original Image:** The raw infrared eye image captured by the tracker.
- **2D Rendering:** Projection of the reconstructed eyeball and pupil region onto the image plane.
- **Eye Tracker + Sphere:** Visualization of the eyeball sphere model aligned with the tracker coordinate system.
- **3D Eyeball:** Fusion of image appearance with the 3D eyeball model for interpretable gaze analysis.
- **3D Mesh:** Geometric mesh reconstruction of the eyeball with anatomical and structural priors.

The video highlights how our model integrates both appearance features and structural constraints, ensuring interpretable and physically consistent gaze estimation.

C METHODOLOGICAL DETAILS

C.1 RENDER SEMANTICS (MGR BRANCH)

We begin by generating the template point clouds for both the pupil and the iris using polar coordinates. The process involves discretizing the angles and radii, and then converting the polar coordinates into 3D Cartesian coordinates.

Generate Angle Grid We generate a set of angles in the range $[0, 2\pi)$ for each batch and frame. This is done by discretizing the angle space into N_{angles} points:

$$\theta_i = \frac{2\pi i}{N_{\text{angles}}}, \quad \text{for } i = 0, 1, 2, \dots, N_{\text{angles}} - 1, \quad (1)$$

this results in an angle grid of shape $[B, N_{\text{angles}}]$, where B is the batch size multiplied by the number of frames.

Generate Radius Grids The radius of the pupil and iris are discretized into N_{radius} points. For the pupil, the radii range from 0 to r_{pupil} , while for the iris, they range from r_{pupil} to r_{iris} . The radii are

generated as:

$$\begin{aligned} r_{\text{pupil}}^i &= \frac{i}{N_{\text{radius}}} r_{\text{pupil}}, \\ r_{\text{iris}}^i &= \frac{i}{N_{\text{radius}}} (r_{\text{iris}} - r_{\text{pupil}}) + r_{\text{pupil}}, \\ \text{for } i &= 0, 1, 2, \dots, N_{\text{radius}} - 1, \end{aligned} \quad (2)$$

this generates two radius grids, one for the pupil and one for the iris, both of shape $[B, N_{\text{radius}}]$.

Convert Polar Coordinates to Cartesian Coordinates. We convert the polar coordinates into 3D Cartesian coordinates for both the pupil and the iris. For each pair of radius r and angle θ , the Cartesian coordinates (x, y, z) are computed as:

$$x = r \cdot \cos(\theta), \quad y = r \cdot \sin(\theta). \quad (3)$$

For the pupil and iris point clouds, the z -coordinate is set to a fixed value, determined by the distance from the camera, L_p , and inverted to place the point clouds in front of the camera:

$$z_{\text{pupil}} = z_{\text{iris}} = -L_p \quad (4)$$

Thus, the final 3D coordinates for the pupil and iris are:

$$\begin{aligned} P_{\text{pupil}} &= (x_{\text{pupil}}, y_{\text{pupil}}, z_{\text{pupil}}), \\ P_{\text{iris}} &= (x_{\text{iris}}, y_{\text{iris}}, z_{\text{iris}}). \end{aligned} \quad (5)$$

The resulting point clouds have shapes $[B, N_{\text{angles}} \times N_{\text{radius}}, 3]$.

C.2 AGE BRANCH DETAILS

A key objective of AGE branch is to train the feature extractor F_{θ_1} under the guidance of spherical fitting, so that gaze features are aligned with the physical definition of eye rotations. However, directly integrating the Isomap (Tenenbaum et al., 2000) algorithm into backpropagation is computationally prohibitive: the time complexity of Isomap is $O(N^2 \log N)$ and the memory complexity is $O(N^2)$, where N denotes the number of samples. Processing hundreds of thousands of gaze features with Isomap is thus infeasible in both time and memory.

Isometric Propagator (IP) To overcome this limitation, we introduce the Isometric Propagator (IP) following previous studies (Bao & Lu, 2024), a lightweight three-layer MLP $IP_{\theta_3}(\cdot)$ that parameterizes the Isomap algorithm. The IP is trained to approximate Isomap embeddings during an initialization phase. Specifically, we freeze the parameters of the pretrained CNN F_{θ_1} and train IP_{θ_3} to regress the Isomap outputs from its input features:

$$\min_{\theta_3} \frac{1}{N} \sum_{i=1}^N L_1(\text{Isomap}(f_i), IP_{\theta_3}(f_i)), \quad (6)$$

where $f_i = F_{\theta_1}(x_i)$ are CNN features and \mathcal{L}_1 denotes the L1 loss.

Retraining the Feature Extractor After training, we freeze the parameters of IP_{θ_3} and replace Isomap with it for training the feature extractor. Sphere-Fitting Training objective is then defined as:

$$\min_{\theta_1} \frac{1}{N} \sum_{i=1}^N L_1(\hat{e}_i, IP_{\theta_3}(F_{\theta_1}(x_i))), \quad (7)$$

where \hat{e}_i denotes the ground-truth embedding derived from spherical fitting. The Isometric Propagator is only used during source-domain training.

Inference At test time, gaze is estimated through Isomap and Spherical Fitting:

$$g_i = SF_{\theta_s}(\text{Isomap}(F_{\theta_1}(x_i))). \quad (8)$$

This training strategy directly optimizes gaze features according to their geometric relation with spherical fitting, ensuring physical interpretability while maintaining computational feasibility.

C.3 TRAINING DETAILS

We use ResNet-18/50 as backbones for fair comparison, followed by our dual-branch decoder. SG-Gaze is trained on NVIDIA A100 GPU, with a batch size of 128. We set the initial weights of projection edge loss λ_{edge} , eyeball center loss λ_{eye}^{center} and pupil center loss λ_{pupil}^{center} to 0.15. The initial weight of that two 3D gaze loss λ_{gaze}^{L2} and $\lambda_{gaze}^{cos-sin}$ are set to 2.5. The training process is terminated at 160 epochs. The Sphere-Fitting Training is 20 epochs, while the IP is trained for 100 epochs on 10000 randomly selected samples.

C.4 VIEW-CONSISTENT REGULARIZATION (VCR)

Rotation parameterization We use the right-handed camera coordinate system and compose the 3D viewpoint rotation as $P(\alpha, \beta, \gamma) = R_z(\gamma)R_y(\beta)R_x(\alpha) \in \text{SO}(3)$, where α, β, γ denote pitch (x-axis), yaw (y-axis) and roll (z-axis), respectively. Unless otherwise stated, angles are in degrees.

Mixture-of-ranges sampling To simultaneously cover near-view perturbations and far-view shifts while keeping training stable, we sample (α, β, γ) from a two-component mixture:

$$(\alpha, \beta, \gamma) \sim (1 - \lambda)\Omega_{\text{loc}} + \lambda\Omega_{\text{glob}},$$

with $\lambda = 0.2$. The local component focuses on small perturbations

$$\Omega_{\text{loc}} = U([-12^\circ, 12^\circ]) \times U([-12^\circ, 12^\circ]) \times U([-6^\circ, 6^\circ]), \quad (9)$$

and the global component sweeps a wider FoV, dataset-aware:

$$\begin{aligned} \Omega_{\text{glob}} = & U([-A_{\text{max}}, A_{\text{max}}]) \times \\ & U([-B_{\text{max}}, B_{\text{max}}]) \times U([-G_{\text{max}}, G_{\text{max}}]). \end{aligned} \quad (10)$$

For UnityEyes pretraining we use $(A_{\text{max}}, B_{\text{max}}, G_{\text{max}}) = (60^\circ, 50^\circ, 15^\circ)$; for TEyeD/LPW fine-tuning we use $(35^\circ, 30^\circ, 10^\circ)$. This setting matches the broader synthetic coverage while avoiding excessive roll in head-mounted real captures. We further adopt a two-stage curriculum: for the first $E=10$ epochs, $\lambda=0$ (local-only), then switch to the above mixture.

AGE branch under rotation (feature-space). Let $f_i \in \mathbb{R}^d$ be the shared-encoder feature of frame I_i . We map the 3D rotation into feature space via the learned linear operator $W \in \mathbb{R}^{d \times 3}$ and its pseudo-inverse W^\dagger :

$$f'_i = f_i P_f, \quad P_f = W P W^\dagger.$$

The analytical decoder Φ_{dec} (shared for original/rotated features) predicts

$$g_i = \Phi_{\text{dec}}(f_i), \quad g'_i = \Phi_{\text{dec}}(f'_i),$$

and VCR enforces rotation-equivariant gaze prediction

$$L_{\text{VCR}}^{\text{gaze}} = w'_{\text{gaze}} \frac{1}{N} \sum_{i=1}^N \|g'_i - P g_i\|_2^2.$$

We do *not* re-encode images after rotation; gradients flow through W , W^\dagger and Φ_{dec} .

MGR branch under rotation (structure-space). From the MGR branch we have canonical pupil/iris point clouds PC_p, PC_i and their camera-space instances P_p^{3D}, P_i^{3D} obtained by the predicted pose $[R|T]$ and then projected by K to P_p^{2D}, P_i^{2D} (see main text). VCR applies the *same* geometric rotation P to the camera-space point clouds (no re-run of MGR):

$$\tilde{P}_p^{3D} = P P_p^{3D}, \quad \tilde{P}_i^{3D} = P P_i^{3D}, \quad \tilde{P}_p^{2D} = K \tilde{P}_p^{3D}, \quad \tilde{P}_i^{2D} = K \tilde{P}_i^{3D}.$$

We enforce 2D semantic edge consistency via nearest-neighbor matching:

$$L_{\text{VCR}}^{\text{edge}} = w'_{\text{proj}} \frac{1}{N} \sum_{i=1}^N \left(\|P_p^{2D} - \tilde{P}_p^{2D}\|_2^2 + \|P_i^{2D} - \tilde{P}_i^{2D}\|_2^2 \right).$$

The total VCR regularization is

$$L_{\text{VCR}} = L_{\text{VCR}}^{\text{gaze}} + L_{\text{VCR}}^{\text{edge}},$$

which used alongside AGE/MGR losses in the joint objective. In practice we share the decoder across original/rotated features (AGE) and reuse the predicted structure for geometric rotation (MGR), which avoids extra backbone passes while enforcing consistent physics across views.

D ADDITIONAL EXPERIMENTS

D.1 EFFECT OF ROTATION ANGLES IN VCR

Motivation Our View-Consistent Regularization (VCR) applies synthetic viewpoint perturbations during training to enforce rotation-equivariant consistency. The choice of rotation ranges may affect the trade-off between local robustness and global generalization. Here we study the influence of different angle ranges on gaze estimation accuracy.

Experimental setup We compared three rotation ranges for yaw, pitch, and roll axes:

- *Small rotation*: yaw $\in [-10^\circ, 10^\circ]$, pitch $\in [-10^\circ, 10^\circ]$, roll $\in [-5^\circ, 5^\circ]$.
- *Medium rotation*: yaw $\in [-20^\circ, 20^\circ]$, pitch $\in [-15^\circ, 15^\circ]$, roll $\in [-10^\circ, 10^\circ]$.
- *Large rotation*: yaw $\in [-40^\circ, 40^\circ]$, pitch $\in [-30^\circ, 30^\circ]$, roll $\in [-15^\circ, 15^\circ]$.

All other settings followed the main training configuration. We report angular gaze error (degrees) on TEyeD (Fuhl et al., 2021) and LPW (Tonsen et al., 2016) benchmarks.

Table 1: Effect of different rotation ranges in VCR on gaze estimation accuracy (angular error, degrees). Medium-range perturbations achieve the best trade-off.

Rotation range	$D_{T_1} \rightarrow D_{T_2} \downarrow$	$D_{T_1} \rightarrow D_S \downarrow$
Small ($\pm 10^\circ / \pm 10^\circ / \pm 5^\circ$)	6.01	3.22
Medium ($\pm 20^\circ / \pm 15^\circ / \pm 10^\circ$)	3.91	1.88
Large ($\pm 40^\circ / \pm 30^\circ / \pm 15^\circ$)	4.65	2.79

Discussion. The results in Table 1 show that overly small perturbations fail to simulate diverse view-points, while excessively large rotations introduce unrealistic samples. Medium-range perturbations strike the best balance, improving cross-view robustness and cross-domain generalization.

Table 2: Quantitative results of individual training and testing of five subjects on TEyeD dataset. After eliminating the influence of kappa angle, the results show that applying more constraints is beneficial to improve the reconstruction accuracy of 3D eyeball model.

Subject	Loss	TEyeD-subset_A				
		3D gaze [°]↓	2D gaze [°]↓	Sem. Iou	2D pupil cent.[px]↓	2D eye cent.[px]↓
subject1	Gaze	1.21	7.59	N/A	N/A	N/A
	Sem. + Gaze + Cent.	0.85	4.66	86.5%	3.11	2.33
subject2	Gaze	1.32	8.50	N/A	N/A	N/A
	Sem. + Gaze + Cent.	0.99	5.21	87.3%	3.45	1.87
subject3	Gaze	1.50	7.05	N/A	N/A	N/A
	Sem. + Gaze + Cent.	1.07	6.33	88.1%	1.22	1.52
subject4	Gaze	1.47	7.32	N/A	N/A	N/A
	Sem. + Gaze + Cent.	1.11	5.08	86.5%	2.51	2.78
subject5	Gaze	1.34	7.74	N/A	N/A	N/A
	Sem. + Gaze + Cent.	1.01	4.57	86.4%	2.71	2.40

D.2 EFFECT OF THE KAPPA ANGLE BETWEEN THE OPTICAL AND VISUAL AXES.

The normalized optical axis g is defined as the vector from the eyeball center o_e to the iris center o_i , $g = \frac{o_i - o_e}{\|o_i - o_e\|}$. We consider g the approximated gaze vector. Note that we did not model the

kappa angle offset between the optical and visual axes. In our previous experiments, we put more supervision on the whole eyeball, such as the center of the eyeball and pupil, as well as the edge of the projection. However, the accuracy has declined. To better understand its impact, we conducted controlled experiments. We trained and tested five subjects separately to eliminate the influence of kappa angle difference among different subjects. The experimental results in Tab 2 highlight that subject-dependent kappa offsets can negatively affect gaze estimation accuracy if ignored. Adding more subject-specific supervision constraints improves consistency of eyeball fitting and yields more reliable 3D gaze estimation.

D.3 SENSITIVITY OF MGR TO 2D EDGE SPARSITY

Motivation. The Model-Guided Reconstruction (MGR) branch is supervised via sparse 2D edge points sampled on the pupil and iris contours. In practice, acquiring dense semantic labels can be expensive; therefore we analyze how the number of sampled 2D edge points K affects reconstruction fidelity and gaze estimation performance. This experiment quantifies the trade-off between annotation cost (semantic sparsity) and final accuracy.

Experimental setup.

- **Backbone & training:** We use the same backbone and training hyperparameters as in the main paper (ResNet-18 / ResNet-50 variants as applicable). The training schedule, optimizer, weight decay and loss weights are kept identical to the main experiments to isolate the effect of K .
- **Point sampling:** For a given K we uniformly sample K_p contour points on the pupil and K_i points on the iris such that $K = K_p + K_i$. By default we keep the pupil : iris ratio the same as in the main paper (e.g., $K_p : K_i = 4 : 1$ if the paper uses 128 pupil vs 32 iris).
- **K values tested:** $K \in \{8, 16, 32, 64, 128, 256\}$.
- **Datasets & evaluation:** Experiments are run on TEyeD- D_{T1} and evaluated on the same test splits used in the main paper. Metrics reported are 3D gaze angular error (degrees), 2D edge reprojection error (mean pixel distance), and semantic IoU.

Losses and weights. We keep the same overall objective as in the main paper:

$$\mathcal{L} = \lambda_g \mathcal{L}_{\text{gaze}} + \lambda_e \mathcal{L}_{\text{edge}} + \lambda_v \mathcal{L}_{\text{vcr}}.$$

When varying K we do *not* change λ_e ; this isolates the effect of the amount of 2D structural supervision. Reported runs keep $\lambda_g, \lambda_e, \lambda_v$ identical to the main experiments.

Table 3: Sensitivity of MGR to number of 2D edge points K . For each K we report 3D gaze angular error (deg, lower better), 2D gaze angular error (deg, lower better), and semantic IoU (% , higher better).

K (total points)	3D gaze ($^\circ$) \downarrow	2D gaze ($^\circ$) \downarrow	Sem. IoU (%) \uparrow
8	2.47 ± 0.08	15.62 ± 0.25	78.5 ± 0.6
16	1.96 ± 0.07	13.24 ± 0.21	83.7 ± 0.5
32	1.64 ± 0.05	11.57 ± 0.18	87.2 ± 0.4
64	1.32 ± 0.04	9.87 ± 0.15	90.4 ± 0.3
128	1.11 ± 0.03	8.82 ± 0.14	92.1 ± 0.3
256	1.29 ± 0.03	9.75 ± 0.13	91.3 ± 0.3

Concluding remark. As shown in Fig. 1, this experiment empirically characterizes the annotation-performance trade-off for MGR and provides guidance for practical deployment: choose $K=128$ that yields near-saturated accuracy (the “knee”), which minimizes labeling cost while retaining reconstruction and gaze performance.

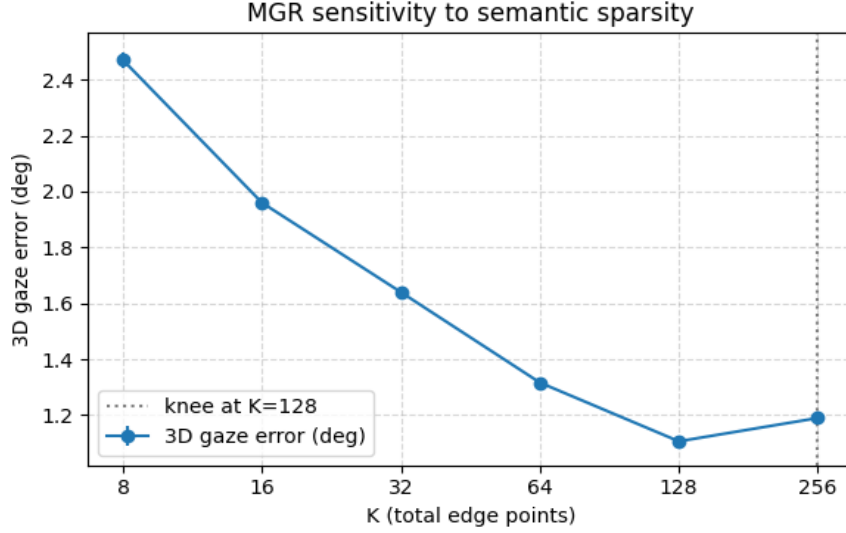


Figure 1: Sensitivity of the MGR branch to the number of 2D edge points K . Increasing K rapidly reduces 3D gaze error up to $K = 128$, beyond which the performance saturates, indicating diminishing returns.

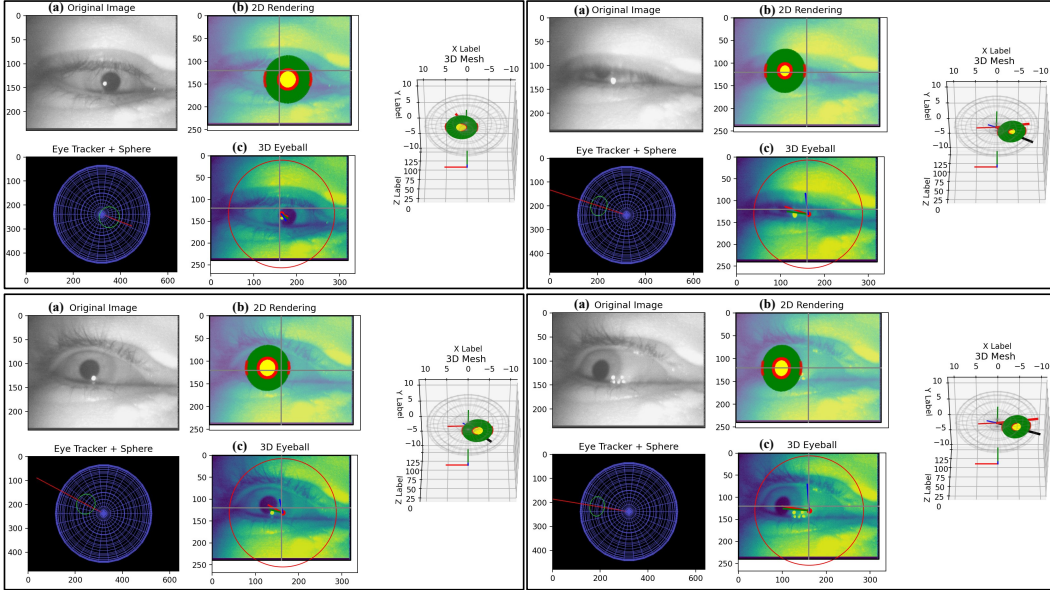


Figure 2: Visualization of gaze estimation under synthetic rotations. Each column shows (a) original image, (b) 2D rendering with VCR, and (c) rotated input with VCR. Red: ground-truth gaze; Blue: prediction without VCR; Green: prediction with VCR. With VCR, the predicted gaze aligns more consistently with the ground truth across different viewpoints, while the iris/pupil edges remain structurally faithful.

D.4 VISUALIZATION OF GAZE CONSISTENCY UNDER ROTATION

Motivation To further illustrate the effectiveness of our View-Consistent Regularization (VCR), we visualize predicted gaze vectors and eyeball projections under different synthetic rotations. This highlights how VCR enforces rotation-equivariant consistency in both appearance and structure.

Visualization setup We apply yaw and pitch perturbations of $\pm 20^\circ$ and project the reconstructed eyeball structures before and after rotation. For each case, we plot:

- Ground-truth gaze vector (red arrow).
- Predicted gaze vector without VCR (blue arrow).
- Predicted gaze vector with VCR (green arrow).
- Corresponding 2D iris/pupil edge projections (overlayed in the image).

Discussion As shown in Fig. 2, models trained without VCR are sensitive to viewpoint changes, causing gaze vectors to drift away from the ground truth and inconsistent iris contours. In contrast, VCR enforces consistent predictions under rotations, leading to both geometrically faithful eyeball reconstructions and improved cross-view gaze alignment.

E LIMITATION AND FUTURE WORK

E.1 LIMITATIONS

Although SG-Gaze demonstrates strong accuracy and cross-domain generalization, several limitations remain that open promising directions for future work.

(1) Subject-specific anatomical variation Our framework currently approximates gaze by aligning the optical axis with the visual axis and does not explicitly model subject-dependent kappa angle offsets. As shown in our supplementary experiments, this simplification may introduce residual bias across individuals. Future work will investigate lightweight calibration strategies or personalized modules to better adapt to subject-specific anatomy.

(2) Dependence on sparse 2D edge supervision The Model-Guided Reconstruction (MGR) branch relies on weak 2D edge labels (pupil and iris contours). While our sensitivity analysis shows that even sparse labels are effective, annotation effort is still required. Extending MGR to leverage unsupervised geometric cues or self-supervised contour discovery could further reduce dependence on human annotation.

(3) Synthetic-to-real domain gaps Although the proposed View-Consistent Regularization (VCR) alleviates domain gaps, our training pipeline still relies on synthetic perturbations that may not capture the full diversity of real-world conditions (e.g., extreme illumination, occlusions, eyeglasses). Incorporating physics-based rendering, domain adaptation, or generative data augmentation may further improve robustness.

(4) Deployment constraints Our method has not yet been fully optimized for resource-constrained AR/VR headsets. Exploring lightweight backbones, pruning, or distillation will be essential to enable real-time gaze tracking on mobile or embedded platforms.

E.2 FUTURE WORK

Building on SG-Gaze, several promising research directions can further align with the broader goals of the ICLR community:

Subject-adaptive representation learning: Develop lightweight calibration modules or meta-learning strategies to account for kappa angle offsets and anatomical variations, advancing personalized yet generalizable models.

Self-supervised structural learning: Move beyond annotated contours by exploiting self-supervised objectives and geometric consistency priors, enabling scalable training on large unlabeled datasets.

Physics-aware domain adaptation: Combine physically-grounded rendering, adversarial domain alignment, and generative augmentation to capture real-world shifts in illumination, occlusion, and device-specific imaging.

Resource-efficient model design: Pursue pruning, quantization, and distillation for low-latency deployment on AR/VR devices, bridging the gap between algorithmic advances and practical on-device applications.

Multi-task and cross-modal extensions: Explore joint learning with related tasks (e.g., iris recognition, user identification, eye-movement behavior analysis), and investigate integration with multi-modal signals such as speech or head motion for richer human-centric modeling.

REFERENCES

- Yiwei Bao and Feng Lu. From feature to gaze: A generalizable replacement of linear layer for gaze estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1409–1418, 2024.
- Wolfgang Fuhl, Gjergji Kasneci, and Enkelejda Kasneci. Teyed: Over 20 million real-world eye images with pupil, eyelid, and iris 2d and 3d segmentations, 2d and 3d landmarks, 3d eyeball, gaze vector, and eye movement types. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 367–375. IEEE, 2021.
- Joshua B Tenenbaum, Vin de Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.
- Marc Tonsen, Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Labelled pupils in the wild: a dataset for studying pupil detection in unconstrained environments. In *Proceedings of the ninth biennial ACM symposium on eye tracking research & applications*, pp. 139–142, 2016.