EyeSeg: An Uncertainty-Aware Eye Segmentation Framework for AR/VR

Zhengyuan Peng[1], **Jianqing Xu**[2], **Shen Li**[3], **Jiazhen Ji**[2], **Yuge Huang**[2]
Jingyun Zhang[2], **Jinmin Li**[4], **Shouhong Ding**[2], **Rizen Guo**[2], **Xin Tan**[♯5] and **Lizhuang Ma**[1,5]
[1]Shanghai Jiao Tong University
[2]Tencent
[3]National University of Singapore
[4]Tsinghua University
[5]East China Normal University

Abstract

Human-machine interaction through augmented reality (AR) and virtual reality (VR) is increasingly prevalent, requiring accurate and efficient gaze estimation which hinges on the accuracy of eye segmentation to enable smooth user experiences. We introduce EyeSeg, a novel eye segmentation framework designed to overcome key challenges that existing approaches struggle with: motion blur, eyelid occlusion, and train-test domain gaps. In these situations, existing models struggle to extract robust features, leading to suboptimal performance. Noting that these challenges can be generally quantified by uncertainty, we design EyeSeg as an uncertainty-aware eye segmentation framework for AR/VR wherein we explicitly model the uncertainties by performing Bayesian uncertainty learning of a posterior under the closed set prior. Theoretically, we prove that a statistic of the learned posterior indicates segmentation uncertainty levels and empirically outperforms existing methods in downstream tasks, such as gaze estimation. EyeSeg outputs an uncertainty score and the segmentation result, weighting and fusing multiple gaze estimates for robustness, which proves to be effective especially under motion blur, eyelid occlusion and cross-domain challenges. Moreover, empirical results suggest that EyeSeg achieves segmentation improvements of MIoU, E1, F1, and ACC surpassing previous approaches.

1 Introduction

In recent years, human-machine interaction, especially augmented reality (AR) and virtual reality (VR), is increasingly being popularized by major smart hardware products. For such products to deliver smooth interaction, it is essential to have a gaze estimation algorithm to capture users' iris location and movement accurately and efficiently. Specifically, the algorithm takes as input a real-time image captured by a camera installed in the equipment and returns the resulting segmentation image that specifies the location of the iris in a pixel-wise manner. This segmentation-based product solution usually confronts three challenges: (i) imagery blurriness: due to eye fast motion, images captured usually exhibit blurriness, which poses difficulty to accurate eye segmentation; (ii) eyelid occlusion; (iii) train-test domain gaps: users may use the equipment in various device configurations or varying usage habits, hence inevitably there exists domain gaps between training data and test data. We demonstrate these challenges in Fig. 4.

Humans typically express their uncertainty when faced with blurry information or when there are biases in domain knowledge; this allows them to make improvements or reject uncertain conclusions in subsequent analyzes. Inspired by this human trait, we propose a novel uncertainty-aware framework for eye segmentation (EyeSeg) designed to resolve the aforementioned challenges. Unlike previous methods [Wei *et al.*, 2021; Czolbe *et al.*, 2021; Sirohi *et al.*, 2023; Fuhl *et al.*, 2024] that rely on indirect or proxy measures of uncertainty, our approach explicitly models the uncertainty of the segmentation results. Specifically, we address the variations of input iris images within the eye regions which form a closed set hereinafter. Compared to open-set, the nature of closed-set conveniently induces a desired prior which a posterior approximates to achieve Bayesian uncertainty learning. Leveraging this prior, we utilize a statistic of the learned posterior to estimate posterior probabilities. Theoretically, we prove that the statistic of the learned posterior reflects the level of uncertainty for segmentation results.

In our framework, beyond the conventional segmentation pipeline, we extract multi-level features from the segmentation network and predict per-pixel variance, which is then used to compute the overall uncertainty of the image. During the testing phase, if the uncertainty exceeds a predefined threshold, the corresponding prediction is discarded. The pipeline is shown in Figure 1. This uncertainty score can be used as a weight to yield a more reliable gaze estimate, especially in the conditions of eyelid occlusion, imagery blurriness and in-domain/cross-domain settings. Moreover, our algorithm operates directly on the detected eye patch, leading to significant improvements in metrics such as Mean Intersection over Union (MIoU), E1, F1, and ACC, while maintaining computational efficiency with only 1.53G FLOPs.

♯ Corresponding Author

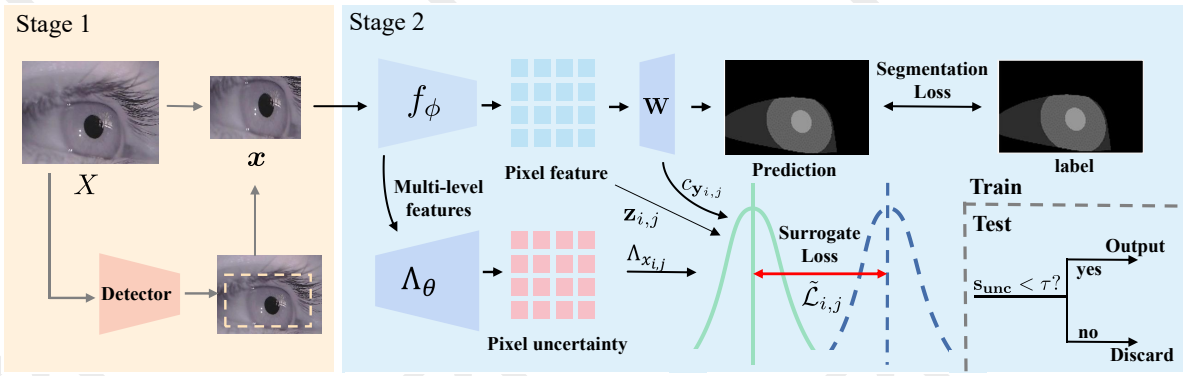

Figure 1: **The procedure of the proposed framework, EyeSeg.** EyeSeg first detects eye regions, then extracts pixel-wise visual features and uncertainties used for Bayesian learning of a posterior which further yields an uncertainty score s_{unc} during test. Here, τ denotes the threshold for decision-making. A detailed description of the process is provided in Algorithm 1 and Algorithm 2.

To sum up, our major contributions are listed Ías follows:

• **(Novelty)** We propose a novel uncertainty-aware eye segmentation framework that resolves common challenges in AR/VR.

• **(Interpretability)** We prove that a statistic of the learned posterior provably reflects the level of uncertainty of segmentation results.

• **(Evaluation)** We conduct extensive experiments on multiple real-world datasets and challenge situations, demonstrating significant improvements in metrics such as MIoU, E1, F1, and ACC with 1.53G FLOPs.

2 Related Work

Eye Segmentation. Semantic segmentation is a fundamental task in computer vision with numerous applications, and there have been significant advancements [Long *et al.*, 2015; Ronneberger *et al.*, 2015; Badrinarayanan *et al.*, 2017; Chen *et al.*, 2014; Chen *et al.*, 2017a; Chen *et al.*, 2017b; Chen *et al.*, 2018; Gong *et al.*, 2021; Xie *et al.*, 2021; Yuan *et al.*, 2023; Tan *et al.*, 2023; Peng *et al.*, 2023; Yuan *et al.*, 2023; Erisen, 2024; Chen *et al.*, 2024; Liu *et al.*, 2024] in recent years. An important application scenario is gaze estimation in AR/VR, where eye segmentation serves as a crucial preprocessing step. Various approaches[Hansen and Pece, 2005; Świrski *et al.*, 2012; Chaudhary *et al.*, 2019; Yiu *et al.*, 2019; Feng *et al.*, 2022; Luo *et al.*, 2020; Wang *et al.*, 2020; Kothari *et al.*, 2021; Biswas and Lescroart, 2023; Fuhl *et al.*, 2024] have been developed, taking into account both the unique characteristics of eye images and the real-time requirements of such applications. Traditional methods[Hansen and Pece, 2005; Świrski *et al.*, 2012] typically employ morphological operations and geometric features to extract pupil regions from eye images. Deep learning-based methods[Chaudhary *et al.*, 2019; Yiu *et al.*, 2019; Feng *et al.*, 2022; Luo *et al.*, 2020; Wang *et al.*, 2020; Kothari *et al.*, 2021; Biswas and Lescroart, 2023], particularly convolutional neural networks, have shown promising results in eye segmentation. These methods leverage the ability of CNNs to learn hierarchical features from large amounts of data, which can be beneficial in handling complex and challenging scenarios. Some CNNs methods[Wang *et al.*, 2020; Kothari *et al.*, 2021; Biswas and Lescroart, 2023] also utilize ellipse fitting to obtain the shapes of the iris and pupil. Unlike these methods mentioned above, our work develops an uncertainty-aware eye segmentation framework which provides theoretically-grounded uncertainty score besides eye segmentation results, offering an *efficient*, *accurate*, and *robust* solution for eye segmentation. Recently, transformer-based methods [Hassan *et al.*, 2022; Wei *et al.*, 2021] have been proposed for eye segmentation. However, due to their high computational complexity, these methods often fail to meet real-time requirements, making them unsuitable for AR/VR applications.

Uncertainty Estimation. Existing methods mainly includes Probabilistic methods [Kendall and Gal, 2017; Czolbe *et al.*, 2021], Evidence Learning[Sirohi *et al.*, 2023], Deterministic Methods [Czolbe *et al.*, 2021; Wei *et al.*, 2021] and Ensemble methods [Czolbe *et al.*, 2021; Landgraf *et al.*, 2024]. In particular, BiTrans [Wei *et al.*, 2021] leverages uncertainty to refine the eye segmentation, but it is not designed for eye segmentation in VR/AR scenes. Our approach belongs to the category of probabilistic methods and exhibits a high degree of interpretability. In contrast to other probabilistic methods, our approach directly models uncertainty in a closed set, where uncertainty is computed only at the subregion level. We leverage uncertainty estimation to reject erroneous outputs, instead of using it solely for improving model performance.

3 Methodology

Our proposed approach, EyeSeg, first detects eye regions, followed by uncertainty-aware eye segmentation. Eye regions provide a closed set, which conveniently induces a desired prior for posterior approximation to achieve uncertainty awareness. The entire approach is illustrated in Fig. 1.

Notation. Throughout the paper, we let (X, Y) be a training pair drawn from a data set, where X denote the raw gray-scale training image and Y the corresponding K-class pixel-level label (i.e. the segmentation map). Note that X and Y are two-dimensional tensors of which $X_{i,j}$ represents the pixel value and $Y_{i,j}$ denotes the segmentation label at position (i, j). Since X's are captured by a head camera, X's contain inference background besides eyes. Hence, $Y_{i,j}$'s reside in an open set that may contain unknown classes.

3.1 Eye Detection

In this step, we aim to exclude interference regions from the image, which typically contains uncontrolled noisy backgrounds. To achieve this, we use object detection techniques to localize eye sockets. By accurately locating the eye socket, we can effectively identify the crucial subregions for eye segmentation. This localization step allows us to narrow down the search space and achieve more efficient and accurate eye segmentation.

Specifically, we first train an eye detection model $d(\cdot)$ that takes X as input and returns a rectangular bounding box (l, t, h, w) that envelops a subregion $\boldsymbol{x}$ which only contains an eye (see the supplement for the specific training procedure). Using the bounding box, the corresponding segmentation label Y is also cropped into a subregion label $\boldsymbol{y}$. All the resulting $\boldsymbol{x}$ and $\boldsymbol{y}$ are resized into $H \times W$. Note that $\boldsymbol{y}_{i,j}$'s now reside in a closed-set $\{0, 1, 2, 3\}$ that corresponds to background, eye, iris, and pupil, respectively. We denote the distribution of the resulting data pair $(\boldsymbol{x}, \boldsymbol{y})$ as $\mathcal{D}$.

3.2 Deterministic Eye Segmentation

We then train a deterministic segmentation network to learn a mapping $f_\phi : \boldsymbol{x} \in \mathbb{R}^{H\times W} \mapsto \boldsymbol{z} \in \mathbb{R}^{H\times W\times D}$, followed by a linear transform and softmax activation on $\boldsymbol{z}$ at each location (i, j):

$$\boldsymbol{p}_{i,j} = \texttt{softmax}(\boldsymbol{W}\boldsymbol{z}_{i,j}), \tag{1}$$

where $\boldsymbol{W}$ is a 4-by-D matrix (D is the dimensionality of $\boldsymbol{z}$) and $\boldsymbol{p}_{i,j} = [\boldsymbol{p}_{i,j}^{(0)}, \boldsymbol{p}_{i,j}^{(1)}, \boldsymbol{p}_{i,j}^{(2)}, \boldsymbol{p}_{i,j}^{(3)}]^T$ is a probability vector, of which each component, $\boldsymbol{p}_{i,j}^{(c)}$, suggests the probability that the corresponding pixel (i, j) belongs to the class c (for $c = 0, 1, 2, 3$). The deterministic segmentation network is trained using the following cross-entropy loss:

$$\min_{\phi,\boldsymbol{W}} \mathcal{L}_{\text{seg}} := \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}} \left[-\sum_{i,j} \boldsymbol{y}_{i,j} \log \boldsymbol{p}_{i,j} \right] \tag{2}$$

During inference, the deterministic segmentation model outputs a segmentation map $\hat{\boldsymbol{y}} \in \mathbb{R}^{H\times W}$, of which the value at each position (i, j) is determined by

$$\hat{\boldsymbol{y}}_{i,j} = \arg\max_c \boldsymbol{p}_{i,j}^{(c)} \tag{3}$$

3.3 Projection Head for Uncertainty-Awareness

The deterministic segmentation model could not handle imagery blurriness and train-test domain gaps, which we find quite ubiquitous in the context of AR/VR products (see Fig. 3 for the examples). In light of this, we build an uncertainty-aware projection header on top of the backbone of the pretrained deterministic segmentation model. This header takes the feature maps of each stage of the segmentation network as input and predicts the covariance of each pixel, thus providing an estimate of the uncertainty associated with the model's predictions. The final output is the uncertainty of an entire image calculated from the pixel-level covariance.

To achieve this, we resort to probabilistic machinery by transforming the deterministic eye segmentation into a probabilistic one. Specifically, we assume that the latent code of a given pixel is no longer a deterministic one but a random variable that follows a Gaussian distribution: $p_\theta(\boldsymbol{z}_{i,j}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}_{i,j}; f_\phi(\boldsymbol{x}, i, j), \Lambda_\theta(\boldsymbol{x}, i, j))$, where $\Lambda_\theta(\cdot)$ is the uncertainty-aware projection head that takes $\boldsymbol{x}$ and an image position (i, j) as input and outputs a diagonal matrix $\Lambda_{\boldsymbol{x}_{i,j}} = \Lambda_\theta(\boldsymbol{x}, i, j)$, which captures uncertainty in the latent space. Note that this projection head can be instantiated by any network with learnable parameters θ.

To learn this posterior distribution $p_\theta(\boldsymbol{z}_{i,j}|\boldsymbol{x}, i, j)$, we notice that the matrix $\boldsymbol{W}$ pretrained in Eq. (1) contains prior knowledge to be utilized: the four rows of $\boldsymbol{W}$ are ideal class template vectors for background, eye, iris, and pupil, respectively. This induces a desired prior q to which p_θ should be regularised. Specifically, the prior q is chosen to be the Dirac delta function, that is, $q(\boldsymbol{z}_{i,j}|\boldsymbol{y}_{i,j}) = \delta(\boldsymbol{z}_{i,j} - \boldsymbol{c}_{\boldsymbol{y}_{i,j}})$, where $\boldsymbol{c}_{\boldsymbol{y}_{i,j}} := \boldsymbol{W}^T \boldsymbol{e}_{\boldsymbol{y}_{i,j}}$. Here the vector $\boldsymbol{e}_{\boldsymbol{y}_{i,j}}$ is a one-hot vector which contains one at the index $\boldsymbol{y}_{i,j}$ and zeros at the rest of indices. Intuitively, the matrix-vector multiplication $\boldsymbol{W}^T \boldsymbol{e}_{\boldsymbol{y}_{i,j}}$ takes the $\boldsymbol{y}_{i,j}{}^{\text{th}}$ row of $\boldsymbol{W}$ and transposes it as a column vector.

To perform regularization, we minimize the cross entropy (CE) between the chosen prior q and the posterior p_θ at all image positions ($i = 1, ..., H, j = 1, ..., W$):

$$\min_\theta \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}} \left[\sum_{i,j} \underbrace{\text{CE}(q(\boldsymbol{z}_{i,j}|\boldsymbol{y}_{i,j}) \,||\, p_\theta(\boldsymbol{z}_{i,j}|\boldsymbol{x}_{i,j}))}_{:=\mathcal{L}_{i,j}} \right] \tag{4}$$

where each summand $\mathcal{L}_{i,j} =$

$$\frac{1}{2}(\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})^T \Lambda_{\boldsymbol{x}_{i,j}}^{-1} (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j}) + \frac{\ln\det(\Lambda_{\boldsymbol{x}_{i,j}})}{2} + \frac{D}{2}\ln 2\pi \tag{5}$$

Here, Eq. (5) is obtained by plugging $q(\boldsymbol{z}_{i,j}|\boldsymbol{y}_{i,j}) = \delta(\boldsymbol{z}_{i,j} - \boldsymbol{c}_{\boldsymbol{y}_{i,j}})$ and $p_\theta(\boldsymbol{z}_{i,j}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}_{i,j}; f_\phi(\boldsymbol{x}, i, j), \Lambda_\theta(\boldsymbol{x}, i, j))$ into Eq. (4).

After optimization, we obtain the optimal θ^* and further, given a test $\tilde{\boldsymbol{x}}$, its optimal $\Lambda^*_{\tilde{\boldsymbol{x}}_{i,j}}$ that captures the dimension-wise latent uncertainty at the pixel location (i, j) of $\tilde{\boldsymbol{x}}$. However, we may expect one summary scalar that quantifies the uncertainty at (i, j). Next, we theoretically show that the desired scalar happens to be the trace of the optimal $\Lambda^*_{\boldsymbol{x}_{i,j}}$.

Theorem 1. *Given an image $\boldsymbol{x}$ and a position (i, j) of interest, the trace of the optimal $\Lambda^*_{\boldsymbol{x}_{i,j}}$ that minimizes each summand of the proposed loss function Eq.* (4) *is equal to the squared Euclidean distance between the latent code $\boldsymbol{z}_{i,j}$ and the corresponding class center $\boldsymbol{c}_{\boldsymbol{y}_{i,j}}$:*

$$\text{tr}(\Lambda^*_{\boldsymbol{x}_{i,j}}) = ||\boldsymbol{z}_{i,j} - \boldsymbol{c}_{\boldsymbol{y}_{i,j}}||_2^2 \tag{6}$$

where

$$\Lambda^*_{\boldsymbol{x}_{i,j}} = \text{diag}(\sigma_1^2, ..., \sigma_D^2) = \arg\min_{\Lambda_{\boldsymbol{x}_{i,j}}} \mathcal{L}_{i,j} \tag{7}$$

Theorem 1 suggests that the closer the latent code $\boldsymbol{z}_{i,j}$ gets to $\boldsymbol{c}_{\boldsymbol{y}_{i,j}}$, the lower the value of $\text{tr}(\Lambda^*_{\boldsymbol{x}_{i,j}})$ becomes. The Euclidean distance can be interpreted as the distance of the prediction at position (i, j) towards its corresponding ideal position in the latent space, and therefore the quantity $\text{tr}(\Lambda^*_{\boldsymbol{x}_{i,j}})$

Algorithm 1: Training Algorithm

Input: The training pairs $(X, Y) \sim \mathcal{D}$; The pretrained detection model $d(\cdot)$.
Output: The deterministic segmentation network f_ϕ, the matrix $\boldsymbol{W}$, the projection head Λ_θ

for $(X, Y) \sim \mathcal{D}$ **do**
  $(l, t, h, w) \leftarrow d(X)$;
  $\boldsymbol{x} \leftarrow$ crop X using (l, t, h, w);
  $\boldsymbol{y} \leftarrow$ crop Y using (l, t, h, w);
$\{\phi, \boldsymbol{W}\} \leftarrow$ Train the eye segmentation network by minimizing $\mathcal{L}_{\text{seg}}$ in Eq. (2);
$\{\theta\} \leftarrow$ Train the projection head by minimizing the surrogate loss in Eq. (9);
return f_ϕ, $\boldsymbol{W}$, Λ_θ

can be seen as the uncertainty for the prediction. In the inference stage, the category of a pixel cannot be obtained in advance, so predicting the category center to which the pixel at (i, j) belongs is an ill-conditioned problem. Our method circumvents this difficulty by estimating an uncertainty $\text{tr}(\Lambda^*_{\boldsymbol{x}_{i,j}})$ that mathematically measures how close a pixel under test is to its unknown class center.

3.4 Optimization Analysis

Solving the original loss function Eq. (4) requires minimizing each summand of Eq. (4) simultaneously. In practice, we find it often converges to suboptimal minima. In this section, we provide a detailed analysis and propose a surrogate loss that can circumvent this optimization difficulty by virtue of Theorem 1.

From Eq. (4), we can find the first derivative of the loss function $\mathcal{L}_{i,j}$:

$$\frac{\partial \mathcal{L}_{i,j}}{\partial \Lambda_{\boldsymbol{x}_{i,j}}} = \frac{1}{2}\Lambda^{-T}_{\boldsymbol{x}_{i,j}}\left(I - (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})(\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})^T\Lambda^{-T}_{\boldsymbol{x}_{i,j}}\right) \tag{8}$$

There are two multiplicative factors that contribute to decreasing the loss derivative. The first factor leads to a trivial solution where the entries of $\Lambda_{\boldsymbol{x}_{i,j}}$ are extremely large. In practice, we observe that the first factor dominates in finding the minimum of Eq. (4) due to random initialization of network parameters in the optimization process. This means that the entire optimization process will probably face optimization issues such as gradient vanishing. In Fig. 2 (left) we showcase the function landscape of the loss Eq. (4) as a function of $\Sigma_{\boldsymbol{x}_{i,j}}$ in the two-dimensional case.

To circumvent the optimization difficulty of loss Eq. (4), we derive a surrogate loss based on Theorem 1. According to Theorem 1, the diagonals are all that we need for uncertainty estimation. Therefore, we only have to have the uncertainty module $\Lambda_\theta(\cdot)$ to output the diagonals and make them approximate the desired quantity that is only available during training, $(\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j}) \odot (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})$:

$$\min_\theta \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\left[\sum_{i,j}\tilde{\mathcal{L}}_{i,j}\right] \tag{9}$$

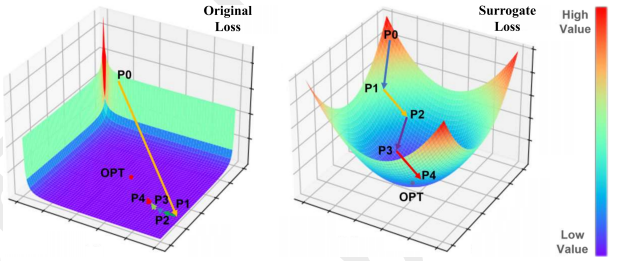

Figure 2: **The function landscapes of the original loss Eq.** (4) **(left) and the surrogate loss Eq.** (9) **(right)**. For visualization, these loss functions are illustrated as functions of two free variables w1, w2 of the covariance $\Lambda_{\boldsymbol{x}_{i,j}}$. We mark an optimization path (**P0** $\rightarrow$ **P4**) and the optimal point (**OPT**) for each. It can be observed that the original loss is prone to optimization issues such as gradient vanishing, whereas the surrogate loss is easy to optimize.

where each summand $\tilde{\mathcal{L}}_{i,j}$ is

$$\tilde{\mathcal{L}}_{i,j} = \left\|\text{diag}(\Lambda_{\boldsymbol{x}_{i,j}}) - (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j}) \odot (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})\right\|_2^2 \tag{10}$$

Note that the function landscape of the surrogate loss Eq. (9) is flat and smooth as compared to the original loss Eq. (4), as shown in Fig. 2. Therefore, the optimization objective is more amenable, as will be shown in our experiments. See Algorithm 1 for the detailed training procedure.

3.5 Inference

After training, we obtain the resulting eye segmentation model f_ϕ, the pretrained matrix $\boldsymbol{W}$ and the projection head Λ_θ. Then, given a raw testing image X, we first use the pretrained eye detection model $d(\cdot)$ to obtain the patch $\boldsymbol{x}$, then use f_ϕ and $\boldsymbol{W}$ to obtain the segmentation result $\hat{\boldsymbol{y}}$. Further, besides the segmentation result, we can obtain the predicted pixel-wise covariance $\Lambda_{\boldsymbol{x}_{i,j}} = \Lambda_\theta(\boldsymbol{x}, i, j)$ at every position (i, j). However, we expect an associated uncertainty score s_{unc} that can summarize the uncertainty of the whole predicted result $\hat{\boldsymbol{y}}$. In this section, we show how to derive this quantity from the predicted pixel-wise covariance.

Algorithm 2: Test Algorithm

Input: The testing image X; The pretrained eye detection model $d(\cdot)$; The eye segmentation model f_ϕ, the pretrained matrix $\boldsymbol{W}$, the projection head Σ_θ.
Output: The predicted segmentation map $\hat{\boldsymbol{y}}$ and its associated uncertainty score s_{unc}

$(l, t, h, w) \leftarrow d(X)$;
$\boldsymbol{x} \leftarrow$ crop X using (l, t, h, w);
$\boldsymbol{z} \leftarrow f_\phi(\boldsymbol{x})$;
for (i, j) **do**
  $\boldsymbol{p}_{i,j} \leftarrow \texttt{softmax}(\boldsymbol{W}\boldsymbol{z}_{i,j})$;
  $\hat{\boldsymbol{y}}_{i,j} \leftarrow \arg\max_c \boldsymbol{p}^{(c)}_{i,j}$;
  $\Lambda_{\boldsymbol{x}_{i,j}} \leftarrow \Lambda_\theta(\boldsymbol{x}, i, j)$
$s_{\text{unc}} \leftarrow \sum_{i,j} \ln\det(\Lambda_{\boldsymbol{x}_{i,j}})$
return $\hat{\boldsymbol{y}}$, s_{unc}

We propose to use the negative logarithm of the posterior density $p(\boldsymbol{z}|\boldsymbol{x})$ as the uncertainty score s_{unc}. The intuition is that the higher the density $p(\boldsymbol{z}|\boldsymbol{x})$ is, the lower the uncertainty score. Specifically, the posterior density can be factorized into:

$$p(\boldsymbol{z}|\boldsymbol{x}) = \prod_{i,j} p(\boldsymbol{z}_{i,j}|\boldsymbol{x}) = \prod_{i,j} \frac{e^{-\frac{1}{2}(\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})^T \Lambda^{*-1}_{\boldsymbol{x}_{i,j}} (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})}}{\sqrt{2\pi \det(\Lambda^*_{\boldsymbol{x}_{i,j}})}} \tag{11}$$

Let $\boldsymbol{v} = (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})$. Then, for the quadratic form

$$\gamma_{i,j} = (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j})^T \Lambda^{*-1}_{\boldsymbol{x}_{i,j}} (\boldsymbol{c}_{\boldsymbol{y}_{i,j}} - \boldsymbol{z}_{i,j}), \tag{12}$$

$$\operatorname{tr}(\gamma_{i,j}) = \operatorname{tr}\left(\boldsymbol{v}^T \begin{pmatrix} \sigma_1^{-2} & & \\ & \ddots & \\ & & \sigma_D^{-2} \end{pmatrix} \boldsymbol{v} \right) \tag{13}$$

$$= \operatorname{tr}\left(\boldsymbol{v}^T \begin{pmatrix} \frac{1}{\boldsymbol{v}_1^2} & & \\ & \ddots & \\ & & \frac{1}{\boldsymbol{v}_D^2} \end{pmatrix} \boldsymbol{v} \right) \tag{14}$$

$$= \operatorname{tr}\left(\boldsymbol{v}\boldsymbol{v}^T \begin{pmatrix} \frac{1}{\boldsymbol{v}_1^2} & & \\ & \ddots & \\ & & \frac{1}{\boldsymbol{v}_D^2} \end{pmatrix} \right) \tag{15}$$

$$= \operatorname{tr}\left(\begin{pmatrix} \boldsymbol{v}_1^2 & \boldsymbol{v}_1\boldsymbol{v}_2 & \\ \boldsymbol{v}_2\boldsymbol{v}_1 & \ddots & \\ & & \boldsymbol{v}_D^2 \end{pmatrix} \begin{pmatrix} \frac{1}{\boldsymbol{v}_1^2} & & \\ & \ddots & \\ & & \frac{1}{\boldsymbol{v}_D^2} \end{pmatrix} \right) \tag{16}$$

$$= \operatorname{tr}\left(\begin{pmatrix} 1 & * & \cdots & * \\ * & 1 & \cdots & * \\ \vdots & \vdots & \ddots & \vdots \\ * & * & \cdots & 1 \end{pmatrix} \right) = D \tag{17}$$

Substituting Eq. (17) into Eq. (11). we can derive the uncertainty estimate for the whole result from pixel-level estimate:

$$p(\boldsymbol{z}|\boldsymbol{x}) \propto \prod_{i,j} \frac{1}{\sqrt{\det(\Lambda^*_{\boldsymbol{x}_{i,j}})}} \tag{18}$$

Therefore, our proposed uncertainty score s_{unc} for the whole segmented result is given by

$$-\ln p(\boldsymbol{z}|\boldsymbol{x}) \propto \sum_{i,j} \ln \det(\Lambda^*_{\boldsymbol{x}_{i,j}}) = s_{\text{unc}} \tag{19}$$

The uncertainty score s_{unc} quantifies the predictive confidence of the model in its segmentation output by mapping the posterior probability to the real number domain $\mathbb{R}$. See Algorithm 2 for the procedure of test algorithm.

4 Experiments

4.1 Datasets

There are several publicly available eye segmentation datasets [Fuhl *et al.*, 2015; Fuhl *et al.*, 2016b; Garbin *et al.*, 2020; Fuhl *et al.*, 2021; Tonsen *et al.*, 2016; Fuhl *et al.*, 2016a; Fuhl *et al.*, 2017] that have been established to aid research and algorithm development. We conduct extensive experiments to evaluate our proposed EyeSeg on multiple widely used datasets, including OpenEDS [Garbin *et al.*, 2019], LPW [Tonsen *et al.*, 2015], Dikablis [Fuhl *et al.*, 2022]. These datasets provide a diverse range of real-world scenarios and variations in imaging conditions. Furthermore, we also curate a meticulously annotated dataset, Else [Fuhl *et al.*, 2016b] to assess the performance of competing methods under more challenging conditions. The annotated labels will be open-sourced. All datasets include four classes: background, eye, iris, and pupil.

4.2 Implementation Details

To increase the diversity of training data, we use conventional data augmentation techniques such as random rotation, translation, scaling, and horizontal flipping. All experimental settings are kept the same for fair comparison. We apply gamma correction to the image during the preprocessing stage.

In the preprocessing step, to localize and identify the regions of eyes in images, we employ an object detection of which its network architecture is similar to YOLOv3 [Redmon and Farhadi, 2018]. Specifically, we employ a pruned, scaled-down version to reduce computational complexity. The network architecture is simplified into a shallower model, with the total computation reduced to 43.13M FLOPs. After this step, all regions of eyes are transformed into 96×96 images to serve as input to the segmentation model.

We utilize the lightweight segmentation network architectures DeepVOG [Yiu *et al.*, 2019] and DenseElNet [Kothari *et al.*, 2021] as the backbones for segmentation, which we refer to as Ours(D) and Ours(E), respectively. The input size of the network is standardized to 96×96, and the output category of network segmentation is set to a closed set including 4 categories (Pupil, Iris, Eye and Background). Due to eye patch extraction, our framework has a much smaller computational overhead than that of using full-images as input under the same segmentation network architecture.

Our uncertainty-aware projection header is implemented using a bottleneck architecture. Specifically, it consists of one downsampling layer and two upsampling layers with skip connections that link corresponding outputs at each layer. By incorporating these skip connections, the projection header can leverage the rich feature representations extracted by the backbone network. The downsampling layer reduces the spatial dimensions of the input, capturing high-level contextual information. The subsequent upsampling layers recover the spatial resolution, enabling fine-grained uncertainty estimation at each pixel. This architecture design ensures that our uncertainty-aware projection header can effectively capture both global and local information from the backbone network, which contributes to the robustness and reliability of uncertainty estimation.

4.3 Baseline Methods

In experiments, we consider two experimental settings: segmentation and uncertainty estimation. In segmentation, our proposed model is compared with existing top-performing

Methods	Else		Dikablis		LPW		OpenEDS		Average	Model FLOPs
	Iris	Pupil	Iris	Pupil	Iris	Pupil	Iris	Pupil		
DeepVOG [Yiu *et al.*, 2019]	93.0	87.0	91.1	94.2	79.7	83.9	N/A	89.1	N/A	3.5G
RITnets [Chaudhary *et al.*, 2019]	93.7	88.7	91.6	94.7	86.7	86.3	91.4	95.0	91.0	16.57G
Pylids [Biswas and Lescroart, 2023]	88.2	82.3	89.8	87.2	85.9	89.0	85.7	88.4	87.1	21.15G
Pisto* [Fuhl *et al.*, 2024]	89.8	83.3	N/A	N/A	N/A	N/A	88.6	87.0	N/A	N/A
DeepLabv3+ [Chen *et al.*, 2018]	93.7	84.4	92.2	94.0	85.7	93.8	98.2	96.7	92.3	39.41G
Ours(D)	95.1	87.6	91.7	94.2	90.1	93.5	98.3	97.3	93.5	**125.85M**
Ours(E)	**96.0**	**90.4**	**93.2**	**95.1**	**91.9**	**95.2**	**98.6**	**97.8**	**94.8**	1.53G

Table 1: Segmentation Results (MIoU) on Various Challenging Benchmarks. Best results are marked in bold. Methods marked with * use the application from the papers due to conditions, without task-specific retraining.

and lightweight eye segmentation methods for AR/VR. We consider ad-hoc eye segmentation methods, including DeepVOG [Yiu *et al.*, 2019], RITNet [Chaudhary *et al.*, 2019], Pisto [Fuhl *et al.*, 2024] and Pylids [Biswas and Lescroart, 2023], with settings consistent with the original training settings in their respective papers. Pylids [Biswas and Lescroart, 2023] is a model for keypoint detection and ellipse fitting. Pisto [Fuhl *et al.*, 2024] executes the executable program provided in the paper. For fair comparison, only results from test datasets not used in training are evaluated. In addition, we also consider a generic segmentation method DeepLabv3+ [Chen *et al.*, 2018] using MobileNet backbone. Notably, it has significantly higher computational complexity than the above methods.

In uncertainty estimation, baseline methods include Ensemble [Czolbe *et al.*, 2021], EvPSNet [Sirohi *et al.*, 2023], Dudes [Landgraf *et al.*, 2024] and BiTrans [Wei *et al.*, 2021]. To ensure a fair comparison, we use DeepVOG as the backbone for all the methods.

4.4 Results

In this section, we demonstrate EyeSeg's performance in terms of two experimental settings: segmentation and uncertainty estimation. Performance is measured using metrics including MIoU (Mean Intersection over Union), E1, F1 and ACC. See complete results in Appendix.

Segmentation. Tab. 1 shows the segmentation results across four datasets. On the OpenEDS dataset, our approach outperforms the state-of-the-art. This improvement is attributed to our method's ability to exclude irrelevant regions, leading to a more efficient and accurate model. These findings suggest the advantages of EyeSeg in eye semantic segmentation.

Additionally, we also evaluate the performance of our method on Else, Dikablis and LPW. It demonstrates that our proposed framework achieves better or promising performance using only a segmentation network with a small amount of computation. In Fig. 4, we present a comparison of the segmentation results from our approach against RITNet and DeepVOG on the Else dataset. As shown in the figure, our method demonstrates higher accuracy and robustness across various challenging scenarios. Besides MIoU, we observe that EyeSeg also exhibits superior performance on E1, F1 and ACC in Appendix.

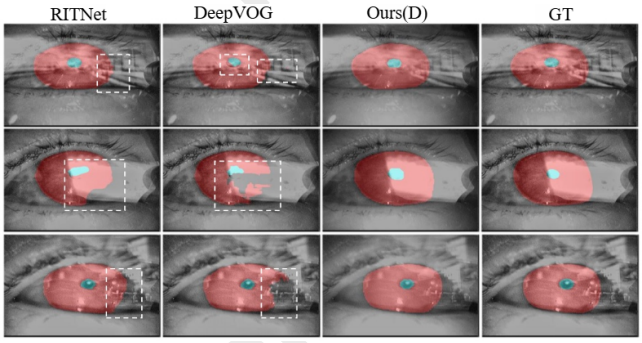

Figure 3: **Segmentation Results.** We visually showcase the segmentation results of our approach compared to RITNet and DeepVOG on the Else dataset. The boxes represent the errors in segmentation outcomes. It serves as a compelling testament to the outstanding precision and resilience exhibited by our approach

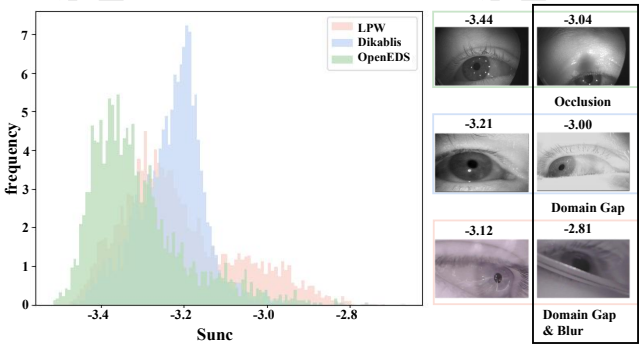

Figure 4: **Uncertainty distribution of different cross-domain datasets.** The model is trained on the Else dataset and tested on others datasets. Different domains have different distribution modes. Our uncertainty score correlates with eye semantics completeness. Compared to normal images, uncertainty is higher in cases of imagery blurriness, eyelid occlusion, and train-test domain gaps.

Uncertainty Estimation. We conduct experiments to evaluate the performance of our uncertainty-aware approach on hard samples in four settings: in-domain, cross-domain, Occlusion and Blur. In the in-domain settings, we evaluate the effects of factors such as image blurriness and noise on model performance. In the cross-domain settings, we evaluate the

Settings	In-domain					Cross-domain				
Thresholds	1%	2%	3%	4%	5%	1%	2%	3%	4%	5%
Ensemble [Czolbe *et al.*, 2021]	87.11	87.14	87.21	87.26	87.33	71.49	71.62	71.73	71.84	71.93
BiTrans [Wei *et al.*, 2021]	86.75	86.80	86.90	86.97	87.10	75.87	75.99	76.10	76.18	76.27
EvPSNet [Sirohi *et al.*, 2023]	87.28	87.28	87.45	87.48	87.48	76.61	76.73	76.79	76.86	76.90
Dudes [Landgraf *et al.*, 2024]	87.65	87.79	87.89	88.01	88.09	74.65	74.74	74.88	74.98	75.06
Ours(D)	**89.32**	**89.39**	**89.46**	**89.68**	**89.72**	**86.56**	**86.76**	**86.89**	**87.02**	**87.15**
Settings	Occlusion					Blur				
Thresholds	1%	2%	3%	4%	5%	1%	2%	3%	4%	5%
Ensemble [Czolbe *et al.*, 2021]	86.50	86.70	86.76	86.93	86.93	84.53	84.93	84.94	85.04	85.06
BiTrans [Wei *et al.*, 2021]	86.16	86.29	86.38	85.48	86.67	83.64	83.66	83.83	83.95	84.20
EvPSNet [Sirohi *et al.*, 2023]	86.72	86.85	86.91	86.91	86.89	84.13	84.21	84.24	84.25	84.20
Dudes [Landgraf *et al.*, 2024]	87.21	87.35	87.43	87.56	87.68	84.96	85.10	85.10	85.20	85.21
Ours(D)	**89.23**	**89.35**	**89.53**	**89.58**	**89.60**	**88.04**	**88.09**	**88.19**	**88.29**	**88.36**

Table 2: Segmentation results (MIoU) after removing images with high uncertainty scores (s_{unc}). Best results are marked in bold. All models are trained and evaluated on the Else dataset, except in the Cross-domain setting, where they are trained on Else and evaluated on OpenEDS.

Ablation I	Thresholds					Ablation II	Thresholds				
	1%	2%	3%	4%	5%		1%	2%	3%	4%	5%
Original loss	86.4	86.5	86.6	86.7	86.7	Entire image	75.5	75.6	75.8	75.9	76.0
Surrogate loss	**86.6**	**86.8**	**86.9**	**87.0**	**87.2**	Ours(D)	**86.6**	**86.8**	**86.9**	**87.0**	**87.2**

Table 3: Ablation study I and II. All models are trained on the Else dataset and evaluated on the OpenEDS dataset.

effects of domain gaps on model performance. In the occlusion setting, we simulate real-world scenarios where parts of the eye are partially covered and assess how well the model can still segment the eye accurately. For the blur setting, we introduce varying levels of motion blur to the test samples to gauge the model's robustness in maintaining segmentation quality under degraded image conditions.

We collect all images from the dataset and rank them according to uncertainty scores generated by different models. With images of large uncertainty scores filtered out, we expect the rest of images to attain good segmentation results. To ensure fairness, all methods are evaluated based on MIoU. As demonstrated in Tab. 2, the images selected by our method show better results than those by other uncertainty-aware methods. This indicates that our proposed uncertainty-aware projection header is more effective for eye segmentation.

In Fig. 4, the uncertainty scores obtained by our method align with human cognition: 1) Our proposed uncertainty scores across cross-domain datasets positively correlate with their similarity to the training dataset Else; 2) Within each dataset, the uncertainty scores generally correlate positively with eye semantics completeness.

4.5 Ablation Study

Study I. We conduct experimental comparisons of the two optimization objectives proposed Eq. (9) in this paper. From the results presented in the Tab. 3, it can be observed that both the optimization objectives yield improvements. The results obtained from each stage of Tab. 3 demonstrate that the surrogate loss function provides a more accurate estimation of the uncertainty associated with the segmentation results.

Study II. We conduct a comprehensive comparison between our method's uncertainty estimation and segmentation performance on both the entire image and sub-regions. As shown in Tab. 3, our method consistently outperforms the other regardless of the proportion of data removed according to the uncertainty scores estimated by our framework.

5 Conclusion

We have proposed an uncertainty-aware eye segmentation framework, EyeSeg, designed to address major challenge in AR/VR: motion blur, eyelid occlusion and train-test domain gaps. We first extract eye patches and then perform uncertainty estimation and segmentation on them. The extracted eye patches satisfy the closed-set condition, reducing the model's difficulty in processing redundant background information and enhancing its robustness for cross-domain data segmentation. The patch extraction allows us to perform an uncertainty estimation Bayesian approach under which a posterior approximates a designed prior induced by the closed set. In addition, we introduce a novel theoretical-grounding approach for uncertainty estimation in this task. Our model surpasses the state-of-the-art methods.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62302167, U23A20343, 62472282, 72192821), Young Elite Scientists Sponsorship Program by CAST (YESS20240780),Shanghai Sailing Program (23YF1410500), Chenguang Program of Shanghai Education Development Foundation and Shanghai Municipal Education Commission (23CGA34).

Contribution Statement

This work was a collaborative effort by all contributing authors. Zhengyuan Peng, Jianqing Xu made equal contributions to this study and are designated as co-first authors. Jianqing Xu and Shen Li acted as project leads, offering initial ideas and guidance throughout the research process. Xin Tan, serving as the corresponding authors, are responsible for all communications related to this manuscript.

References

[Badrinarayanan *et al.*, 2017] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.

[Biswas and Lescroart, 2023] Arnab Biswas and Mark D Lescroart. A framework for generalizable neural networks for robust estimation of eyelids and pupils. *Behavior Research Methods*, pages 1–23, 2023.

[Chaudhary *et al.*, 2019] Aayush K Chaudhary, Rakshit Kothari, Manoj Acharya, Shusil Dangi, Nitinraj Nair, Reynold Bailey, Christopher Kanan, Gabriel Diaz, and Jeff B Pelz. Ritnet: Real-time semantic segmentation of the eye for gaze tracking. In *2019 IEEE/CVF International Conference on Computer Vision Workshop*, pages 3698–3702. IEEE, 2019.

[Chen *et al.*, 2014] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

[Chen *et al.*, 2017a] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017.

[Chen *et al.*, 2017b] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[Chen *et al.*, 2018] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 801–818, 2018.

[Chen *et al.*, 2024] Yujun Chen, Xin Tan, Zhizhong Zhang, Yanyun Qu, and Yuan Xie. Beyond the label itself: Latent labels enhance semi-supervised point cloud panoptic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1245–1253, 2024.

[Czolbe *et al.*, 2021] Steffen Czolbe, Kasra Arnavaz, Oswin Krause, and Aasa Feragen. Is segmentation uncertainty useful? In *Information Processing in Medical Imaging: 27th International Conference, IPMI 2021, Virtual Event, June 28–June 30, 2021, Proceedings 27*, pages 715–726. Springer, 2021.

[Erisen, 2024] Serdar Erisen. Sernet-former: Semantic segmentation by efficient residual network with attention-boosting gates and attention-fusion networks. *arXiv preprint arXiv:2401.15741*, 2024.

[Feng *et al.*, 2022] Yu Feng, Nathan Goulding-Hotta, Asif Khan, Hans Reyserhove, and Yuhao Zhu. Real-time gaze tracking with event-driven eye segmentation. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces*, pages 399–408. IEEE, 2022.

[Fuhl *et al.*, 2015] Wolfgang Fuhl, Thomas Kübler, Katrin Sippel, Wolfgang Rosenstiel, and Enkelejda Kasneci. Excuse: Robust pupil detection in real-world scenarios. In *Computer Analysis of Images and Patterns*, pages 39–51. Springer, 2015.

[Fuhl *et al.*, 2016a] Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, and Enkelejda Kasneci. Pupilnet: Convolutional neural networks for robust pupil detection. *arXiv preprint arXiv:1601.04902*, 2016.

[Fuhl *et al.*, 2016b] Wolfgang Fuhl, Thiago C Santini, Thomas Kübler, and Enkelejda Kasneci. Else: Ellipse selection for robust pupil detection in real-world environments. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*, pages 123–130, 2016.

[Fuhl *et al.*, 2017] Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, Wolfgang Rosenstiel, and Enkelejda Kasneci. Pupilnet v2. 0: Convolutional neural networks for cpu based real time robust pupil detection. *arXiv preprint arXiv:1711.00112*, 2017.

[Fuhl *et al.*, 2021] 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 *IEEE International Symposium on Mixed and Augmented Reality*, pages 367–375. IEEE, 2021.

[Fuhl *et al.*, 2022] 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, 2022.

[Fuhl *et al.*, 2024] Wolfgang Fuhl, Daniel Weber, and Shahram Eivazi. Pistol: Pupil invisible supportive tool in the wild. *SN Computer Science*, 5(3):276, 2024.

[Garbin *et al.*, 2019] Stephan J. Garbin, Yiru Shen, Immo Schuetz, Robert Cavin, Gregory Hughes, and Sachin S. Talathi. Openeds: Open eye dataset. *CoRR*, abs/1905.03702, 2019.

[Garbin *et al.*, 2020] Stephan Joachim Garbin, Oleg Komogortsev, Robert Cavin, Gregory Hughes, Yiru Shen, Immo Schuetz, and Sachin S Talathi. Dataset for eye tracking on a virtual reality platform. In *ACM Symposium*

on Eye Tracking Research and Applications, pages 1–10, 2020.

[Gong *et al.*, 2021] Jingyu Gong, Jiachen Xu, Xin Tan, Jie Zhou, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Boundary-aware geometric encoding for semantic segmentation of point clouds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1424–1432, 2021.

[Hansen and Pece, 2005] Dan Witzner Hansen and Arthur EC Pece. Eye tracking in the wild. *Computer Vision and Image Understanding*, 98(1):155–181, 2005.

[Hassan *et al.*, 2022] Bilal Hassan, Taimur Hassan, Ramsha Ahmed, Naoufel Werghi, and Jorge Dias. Sipformer: Segmentation of multiocular biometric traits with transformers. *IEEE Transactions on Instrumentation and Measurement*, 72:1–14, 2022.

[Kendall and Gal, 2017] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Neural Information Processing Systems*, 30, 2017.

[Kothari *et al.*, 2021] Rakshit S Kothari, Aayush K Chaudhary, Reynold J Bailey, Jeff B Pelz, and Gabriel J Diaz. Ellseg: An ellipse segmentation framework for robust gaze tracking. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2757–2767, 2021.

[Landgraf *et al.*, 2024] Steven Landgraf, Kira Wursthorn, Markus Hillemann, and Markus Ulrich. Dudes: Deep uncertainty distillation using ensembles for semantic segmentation. *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 92(2):101–114, 2024.

[Liu *et al.*, 2024] Chang Liu, Xudong Jiang, and Henghui Ding. Primitivenet: decomposing the global constraints for referring segmentation. *Visual Intelligence*, 2(1):16, 2024.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[Luo *et al.*, 2020] Bingnan Luo, Jie Shen, Shiyang Cheng, Yujiang Wang, and Maja Pantic. Shape constrained network for eye segmentation in the wild. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1952–1960, 2020.

[Peng *et al.*, 2023] Zhengyuan Peng, Qijian Tian, Jianqing Xu, Yizhang Jin, Xuequan Lu, Xin Tan, Yuan Xie, and Lizhuang Ma. Generalized category discovery in semantic segmentation. *arXiv preprint arXiv:2311.11525*, 2023.

[Redmon and Farhadi, 2018] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[Ronneberger *et al.*, 2015] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer Assisted Intervention*, 2015.

[Sirohi *et al.*, 2023] Kshitij Sirohi, Sajad Marvi, Daniel Büscher, and Wolfram Burgard. Uncertainty-aware panoptic segmentation. *IEEE Robotics and Automation Letters*, 8(5):2629–2636, 2023.

[Świrski *et al.*, 2012] Lech Świrski, Andreas Bulling, and Neil Dodgson. Robust real-time pupil tracking in highly off-axis images. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, pages 173–176, 2012.

[Tan *et al.*, 2023] Xin Tan, Qihang Ma, Jingyu Gong, Jiachen Xu, Zhizhong Zhang, Haichuan Song, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Positive-negative receptive field reasoning for omni-supervised 3d segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):15328–15344, 2023.

[Tonsen *et al.*, 2015] Marc Tonsen, Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Labeled pupils in the wild: A dataset for studying pupil detection in unconstrained environments. *CoRR*, abs/1511.05768, 2015.

[Tonsen *et al.*, 2016] 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*, pages 139–142, 2016.

[Wang *et al.*, 2020] Caiyong Wang, Jawad Muhammad, Yunlong Wang, Zhaofeng He, and Zhenan Sun. Towards complete and accurate iris segmentation using deep multitask attention network for non-cooperative iris recognition. *IEEE Transactions on information forensics and security*, 15:2944–2959, 2020.

[Wei *et al.*, 2021] Jianze Wei, Huaibo Huang, Muyi Sun, Yunlong Wang, Min Ren, Ran He, and Zhenan Sun. Toward accurate and reliable iris segmentation using uncertainty learning. *arXiv preprint arXiv:2110.10334*, 2021.

[Xie *et al.*, 2021] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.

[Yiu *et al.*, 2019] Yuk-Hoi Yiu, Moustafa Aboulatta, Theresa Raiser, Leoni Ophey, Virginia L Flanagin, Peter zu Eulenburg, and Seyed-Ahmad Ahmadi. Deepvog: Open-source pupil segmentation and gaze estimation in neuroscience using deep learning. *Journal of Neuroscience Methods*, 2019.

[Yuan *et al.*, 2023] Li Yuan, Xinyi Liu, Jiannan Yu, and Yanfeng Li. A full-set tooth segmentation model based on improved pointnet++. *Visual Intelligence*, 1(1):21, 2023.