EllSeg: An Ellipse Segmentation Framework for Robust Gaze Tracking

Rakshit S. Kothari*, Aayush K. Chaudhary*, Reynold J. Bailey, Jeff B. Pelz and Gabriel J. Diaz

Abstract—Ellipse fitting, an essential component in pupil or iris tracking based video oculography, is performed on previously segmented eye parts generated using various computer vision techniques. Several factors, such as occlusions due to eyelid shape, camera position or eyelashes, frequently break ellipse fitting algorithms that rely on well-defined pupil or iris edge segments. In this work, we propose training a convolutional neural network to directly segment entire elliptical structures and demonstrate that such a framework is robust to occlusions and offers superior pupil and iris tracking performance (at least 10% and 24% increase in pupil and iris center detection rate respectively within a two-pixel error margin) compared to using standard eye parts segmentation for multiple publicly available synthetic segmentation datasets.

Index Terms—Head mounted eye-tracking, ellipse fitting, eye-segmentation, AR/VR

1 Introduction

There is great potential for the use of eye tracking in augmented and virtual reality (AR/VR) displays both as a means for user interaction, and for gaze-dependent rendering techniques that can both increase visual fidelity [1] while also lowering computational overhead [2]. Contemporary methods for eye tracking in VR and AR build upon techniques established in the context of head-mounted video-oculography, which involve the use of one or more infrared light sources placed next to infrared *eye* cameras. These eye cameras are pointed towards each of the wearer's eyes while a third camera, referred to as the scene camera, points away from the wearer to capture the environment being observed [3]. Existing solutions extract gaze descriptive features such as pupil center [4–9], pupil ellipse [10–14], iris ellipse [15–17], or track iridial features [18, 19]. These solutions vary in algorithmic complexity, latency, and computational power requirements. Extracted features are then correlated to a measure of gaze using calibration routines [20–22], which compensate for person-specific physiological differences.

Despite many recent advances in eye-tracking technology [23], three factors continue to adversely impact the performance of eye-tracking algorithms: 1) reflections from the surroundings and from intervening optics, 2) occlusions due to eyelashes, eyelid shape, or camera placement and 3) small shifts of the eye-tracker position caused due to slippage [24]. Pupil detection algorithms such as ExCuSe [4] and PuRe [8] which rely on hand-crafted features are particularly susceptible to stray reflections (unanticipated patterns on eye imagery) and occlusion of descriptive gaze regions (such as eyelid covering the pupil or iris). Recent appearance-based methods based on Convolutional Neural Networks (CNNs) are better able to extract reasonably reliable gaze features despite the presence of reflections [25] or occlusions [26]. Additionally, for head-mounted eye-tracking systems, the degradation of gaze estimate accuracy over time due to slippage [27] can be minimized by estimating the 3D eyeball center of rotation [28] (loosely referred at as an 'eyeball fit'). Estimating the precise physiology of the human eye is a complicated process and computationally intractable [29]. By making certain simplifying assumptions [30] about the human eye and its geometrical constraints, an estimate of a *reduced* optical eyeball model can be obtained from 2D pupil [11, 12, 31, 32] or iris [15–17] elliptical fits. These elliptical fits are derived from identified pupil and iris segments or outline [33]. Efforts by Chaudhary *et al*. [25] and Wu *et al*. [34] demonstrate that CNNs can precisely segment eye images into its constituent parts, *i.e*., the pupil, iris, sclera and background skin regions.

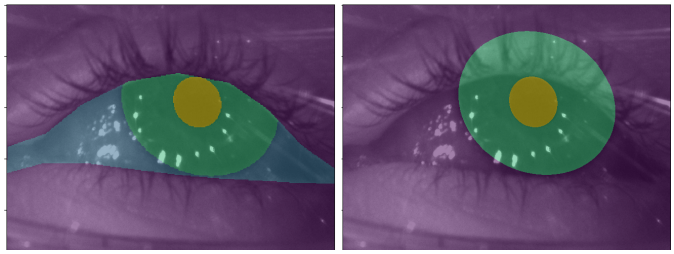

Fig. 1. *PartSeg vs EllSeg*. Left: A *Four-class* eye part segmentation at the pixel-level (i.e. PartSeg) produces labelled pupil (yellow), iris (green), sclera (blue) and background (purple) classes. Right: The EllSeg (three-class) modification produces labelled pupil (yellow) and iris (green) elliptical regions and the rest is marked as background (purple).

In this work, we show that partially occluded pupil or iris regions can result in imprecise or degenerate elliptical fits. To mitigate this, we provide a solution, called *EllSeg*, which is made robust to occlusion by training CNNs to predict entire elliptical eye regions (the full pupil and the full iris) along with the remaining background, as opposed to the standard visible eye-parts segmentation (PartSeg) (see Figure 1). Additionally, we demonstrate that this approach enables us to train segmentation-based CNN architectures directly on datasets wherein only the pupil centers are available [5, 6, 35], allowing us to combine eye parts segmentation and pupil center estimation into a common framework.

The summary of our contributions are as follows:

1. We propose EllSeg, a framework that can be utilized with any encoder-decoder architecture for pupil and iris ellipse segmentation. EllSeg enables prediction of the pupil and iris as full elliptical structures despite the presence of occlusions.

2. To establish the utility of our methodology, we rigorously test our proposed 3-class ellipse segmentation framework using three network architectures, a modified Dense Fully Connected Network [36] (referred as DenseElNet), RITnet [25] and DeepVOG [14]. Performance is benchmarked with well defined train and test splits on multiple datasets, including some which are limited to labelled pupil centers only.

2 Related work

This work is primarily based on the observation that CNNs can identify which category a pixel belongs to despite conflicting appearance

• *All authors are with the Rochester Institute of Technology.*
* *R.S.Kothari and A.K. Chaudhary contributed equally to this work.*

Manuscript received 9 Sept. 2020; revised 15 Dec. 2020; accepted 8 Jan. 2021. Date of publication 29 Mar. 2021; date of current version 7 Apr. 2021. Digital Object Identifier no. 10.1109/TVCG.2021.3067765

(*e.g.* accurately predicting a pixel as belonging to the pupil despite being occluded by eyelids or glasses). Successful segmentation in the presence of ambiguous appearance indicates that a CNN can reason over a wide range of inter-pixel spatial relationships while precise segmentation boundaries indicate successful utilization of fine-grained, high-frequency content observed in local neighborhoods. This ability to capture local information with a global context is achieved by repeatedly pooling intermediate outputs of convolutional operations within a neural network [37]. While numerous architectures can produce a "one-to-one" mapping between an image pixel and its segmentation output class, specific architectures rely on encoding an input image to low dimensional representation followed by decoding and up-sampling to a segmentation map - aptly named encoder-decoder architectures.

Researchers have demonstrated promising results using encoder-decoder architectures for image segmentation. For example, Chaudhary *et al*. [25] proposed RITnet, a lightweight architecture which leverages feature reuse and fixed channel size to maintain low model complexity while demonstrating state of the art performance on the OpenEDS dataset [38]. In this work, we designed our own encoder-decoder architecture called *DenseElNet* which incorporates the dense block proposed by RITnet while leveraging residual connections across each block as proposed by Jegou *et al*. [36]. This ensures a healthy gradient flow and faster convergence while mitigating the vanishing gradient problem [39, 40]. Similar to common encoder-decoder architectures, DenseElNet reduces the spatial extent of its input image but increases the channel size. Note that DenseElNet does not offer any particular novelty over existing encoder-decoder architectures. It is simply being used to facilitate testing of our EllSeg framework.

The primary purpose of eye image segmentation, in the context of gaze estimation, is to produce reliable ellipse fits. The DeepVOG framework by Yiu *et al*. [41] utilizes the U-net architecture to segment the pupil followed by an out-of-network ellipse fitting procedure to generate a 3D model using the "two circles" approach [12,42]. A limitation of their approach is that they segment the pupil based solely on appearance which would likely suffer from occlusion as described previously. Fuhl *et al*. [43] demonstrated that ellipse parameters can be regressed using the bottleneck representation of an input image. However they do not report any metrics for ellipse fit quality. Wu *et al*. [34] leverage multiple decoders to segment an image and estimate 2D cornea and pupil center. Multiple decoders may increase computational requirements and introduce bottlenecks in the pipeline by operating on redundant information. In contrast, we show that the iris and pupil ellipse can be generated using a single encoder-decoder forward pass.

3 Methodology

Figure 2 highlights the EllSeg framework on any generic encoder-decoder (E-D) architecture. First, an input image $I \subset \mathbb{R}$ is passed through an encoder to produce a bottleneck representation Z such that $Z = E(I)$. In our implementation of DenseElNet, I is down-sampled four times by a factor 2 at the bottleneck layer. Subsequently, the network segmentation output O is given by $O = D(Z)$ and consists of three channels (background O_{bg}, iris O_{ir} and pupil O_{pl} output maps). Note that the segmentation outputs are also used to derive pupil and iris ellipse centers. The pupil and iris centers, along with the remaining ellipse parameters (axes and orientation), are also regressed from this bottleneck representation Z using a series of convolutional layers followed by a flattening operation and mapped to a ten-dimensional output (5 parameters for both the iris and pupil ellipses). Please refer to Figure 3 for the ellipse regression module architecture. We test the effectiveness of EllSeg framework on three architectures, DenseElNet (2.18M parameters), RITnet (0.25M parameters), and DeepVOG (3.71M parameters). Note that the regression module is trained alongside the entire network in an End-to-end fashion.

3.1 Ellipse center

The center of any convex shape can be described as a weighted summation of its spatial extent (see Equation 1). In this context, *spatial extent* refers to all possible pixel coordinates while *weight* refers to the probability estimate of a pixel being within the convex structure.

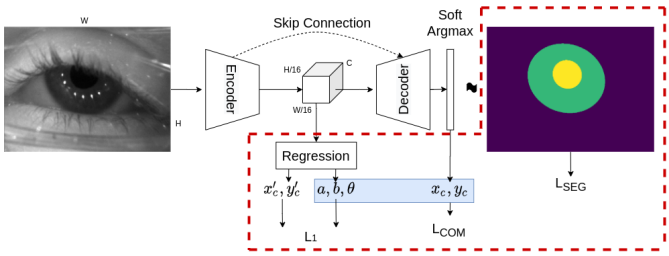

Fig. 2. Proposed EllSeg framework (region enclosed by red dotted line) builds upon existing CNN-based approaches to facilitate the simultaneous segmentation and ellipse prediction for both iris and pupil regions. The resulting ellipse parameters are highlighted in the blue box.

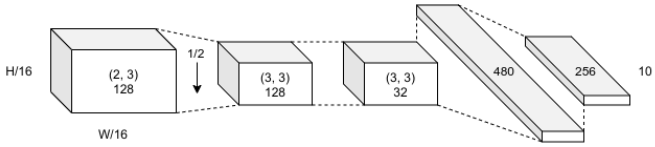

Fig. 3. Regression module architecture. The ↓ signifies average pooling to 1/2 the resolution. Tensors are flattened after three convolutional layers and passed through two linear layers before regressing 10 values (5 ellipse parameters for pupil and iris each).

$$x_c^k, y_c^k = \sum_{i=1}^{W}\sum_{j=1}^{H} p^k_{<i,j>} x, \quad \sum_{i=1}^{W}\sum_{j=1}^{H} p^k_{<i,j>} y, \qquad p^k_{<i,j>} \subset \mathbb{R} \tag{1}$$

Here, x_c^k and y_c^k correspond to the center of a particular feature class k (such as pupil). The iterators i and j span across the width W and height H of an image. The probability values p^k for each pixel are derived after a scaled, spatial softmax operation [44]:

$$p^k = \frac{\exp(\beta O^k_{<i,j>})}{\sum_{i,j=1}^{W,H} \exp(\beta O^k_{<i,j>})} \tag{2}$$

Here, β is a control parameter (also known as temperature [45]), which scales network output around the largest value. We empirically set β as 4. This formulation of ellipse center gives rise to several advantages offered by EllSeg over PartSeg discussed in Section 3.3.2 and Section 6.3.

While one may trivially estimate the pupil center in this manner, deriving the iris center is not straightforward due to its placement *within* the pupil. One alternative is to sum the pupil and iris activation maps before spatial softmax. However, this incorrectly results in the predicted pupil and iris sharing the same 2D center which is physiologically improbable as the pupil is not usually perfectly centered within the iris [46]. Instead, we propose leveraging the background class to predict the iris center in our 3 class segmentation framework. Encoder-decoder architectures have shown to perform exceedingly well at identifying "background" class pixels (see Supplementary Table 1 in Nair *et al*. [47] and Table 2 in Wu *et al*. [34]). To derive the iris center, we negate the background class output map in Equation 2, a modification which subsequently leads to an inverted peak at the predicted iris center location. This inversion ensures the background probability scores do not affect segmentation based loss functions (see Section 6.3.1).

3.2 Ellipse axis and orientation

The bottleneck representation Z is a low dimensional latent representation of the input image. This convenient representation enables us to regress parameters such as the ellipse axis and orientation (we use L_1 loss in our implementation). Experiments revealed that regressing the pupil and iris centers does not offer sub-pixel accuracy (see Section 6.4)

as opposed to deriving them from segmentation output as described in the previous section.

3.3 Loss functions

3.3.1 Segmentation losses $\mathscr{L}_{SEG}$

In the EllSeg framework, the network output O is primarily used to segment an eye image into pupil and iris ellipses, and the background (which includes scleral regions). To train such an architecture, we use the combination of loss functions proposed in RITnet [18]. This strategy involves using a weighted combination of four loss functions; cross-entropy loss, $\mathscr{L}_{CEL}$, generalized dice loss [48] $\mathscr{L}_{GDL}$, boundary aware loss $\mathscr{L}_{BAL}$ and surface loss [49] $\mathscr{L}_{SL}$.

The total loss $\mathscr{L}$ is given by a weighted combination of these losses as $\mathscr{L}_{SEG} = \mathscr{L}_{CEL}(\lambda_1 + \lambda_2\mathscr{L}_{BAL}) + \lambda_3\mathscr{L}_{GDL} + \lambda_4\mathscr{L}_{SL}$. In our experiments, we used $\lambda_1 = 1$, $\lambda_2 = 20$, $\lambda_3 = (1-\alpha)$ and $\lambda_4 = \alpha$, where $\alpha = epoch/M$ and M is the number of epochs.

3.3.2 Center of Mass loss $\mathscr{L}_{COM}$

The L_1 loss function is used to formulate an error function between the center of mass, *i.e.*, the pupil and iris ellipse centers from the segmentation output maps, to their respective ground-truth centers. This enables us to leverage datasets such as ExCuSe [4] , ElSe [5], PupilNet [6] and LPW [35] in a segmentation framework where only the ground-truth pupil center is available. Note that COM L_1 loss (henceforth referred to as $\mathscr{L}_{COM}$ loss) does not impede segmentation loss functions, but instead conditions the network output to jointly satisfy all loss functions. This results in the characteristic peaks observed in Section 6.3.1. The inversion of the background class results in an inverted peak at the iris center location.

4 Datasets

Combining segmentation and $\mathscr{L}_{COM}$ losses allows the EllSeg framework to train CNNs on a large number of datasets (to the best of our knowledge, it enables the inclusion of all publicly available near-eye datasets). To demonstrate the utility of EllSeg, we choose the following datasets for our experiments: NVGaze [9], OpenEDS [38], RITEyes, ElSe [5], ExCuSe [4], PupilNet [50] and LPW [35]. The ElSe and ExCuSe datasets are combined (also referred to as Fuhl) due to similar environment and eyetracker . For more details about each dataset, available ground-truth modality, and train/test splits, please refer to Table 1. Note that we specifically leverage the S-General dataset from the RIT-Eyes framework [47] as it offers wide spatial distribution of *eye* camera position.

4.1 Groundtruth ellipse fits

To obtain groundtruth pupil and iris ellipse fits from the selected datasets, pupil and limbus edges are extracted from groundtruth segmentation masks using a canny edge detector. To ensure subpixel accuracy, we consider edge pixels in the inverted mask as well. Edge pixels which satisfy pupil-iris (*i.e.*, no neighboring sclera or background pixel) or limbus (*i.e.*, no neighboring pupil or background pixel) conditions are used to determine ellipse parameters using the ElliFit algorithm [51] (see Figure 4). Random Sample Consensus (RANSAC) [52] is employed to remove outliers with residuals higher than 5×10^{-3}, an empirically derived threshold . While datasets such as RITEyes and NVGaze directly offer EllSeg compatible groundtruth semantic masks, synthetic masks for OpenEDS were generated based on elliptical fits. Images without valid pupil or iris fits (117 out of 11319) were discarded from all subsequent analysis.

5 Experiments and Hypothesis

We rigorously test various hypotheses to validate the efficacy of our proposed methodology in the field of eye-tracking. In the first experiment (Section 6.1), we benchmark the segmentation performance of our network, DenseElNet, on the standard PartSeg framework. Comparable or superior performance on the PartSeg task will validate DenseElNet. In the second experiment (Section 6.2), we test whether the EllSeg framework improves the detection of both pupil and iris estimates over its PartSeg counterpart. Finally, in the third experiment (Section 6.3), we compare the results of regressing elliptical parameters in the EllSeg framework to those found when estimating the ellipse parameters using RANSAC. This experiment will test whether reliable and differentiable ellipses can be directly estimated in an encoder-decoder architecture. Summary of all the experiments can be found in Figure 5.

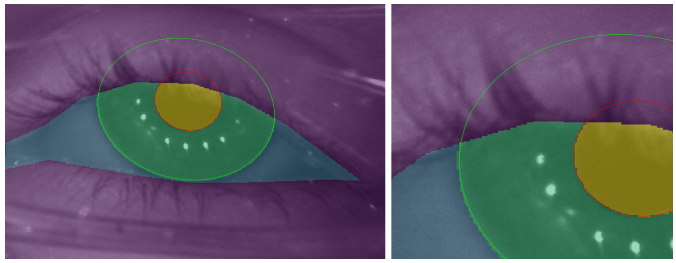

Fig. 4. Ellipse fitting quality on ground truth PartSeg masks. These fits are further used to generate EllSeg masks for the OpenEDS dataset.

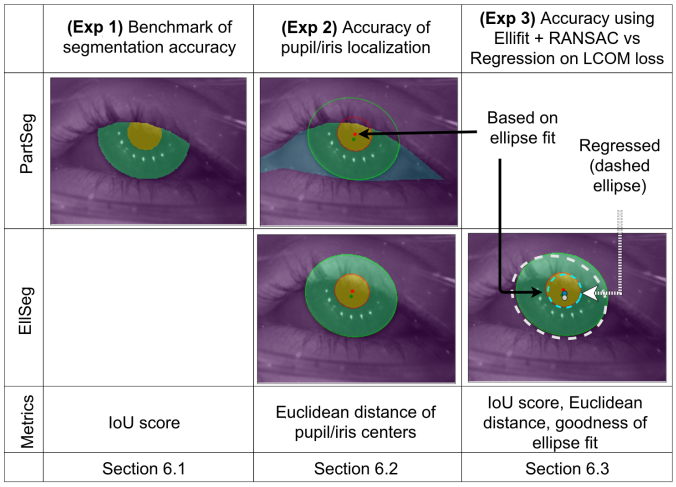

Fig. 5. Summary of all experiments described in following sections (Center estimates are best viewed on screen).

5.1 Training

To ensure fair comparison, all CNN architectures are trained and evaluated with identical train/validation/test splits. The development set is divided into a 80/20% train/validation split. Sample selection is stratified based on binned 2D pupil center position and subsets present within each dataset (see Table 1). This approach ensures that biases introduced due to sampling are minimized while maintaining similar statistical distributions across training and validation sets. Bins with fewer than five images are automatically discarded. All architectures are trained using ADAM optimization [53] on a batch of 48 images at 320x240 resolution with a learning rate of 5×10^{-4} on an NVIDIA V100 GPU.

During training, all models were evaluated with the metric: $[4 + \text{mIoU} - 0.0025(d_p + d_i) - (\theta_p + \theta_i)/90^\circ]$, where mIoU corresponds to the mean intersection over union (IoU) [54] score which quantifies segmentation performance, d_p & d_i are the distances between pupil and iris centers from their groundtruth values in pixels, and θ_p & θ_i are the angular error between the predicted and groundtruth ellipse orientations in degrees. If no improvement above 10^{-3} was observed on this metric for ten consecutive epochs, then a network's parameters were deemed converged. The learning rate was reduced by a factor of ten if no improvements were identified over five epochs. To reduce training time and ensure stable training on pupil-center-only datasets, all models

Table 1. Summary of datasets. ↑ and ↓ correspond to up and down sampling respectively. OpenEDS image crops are extracted around the scleral center followed by up-sampling. Note that images without valid pupil and iris fits are discarded (see Section 4).

Dataset	Resolution	Train subset	Test subset	Groundtruth included	Image Count (train, test)	Preprocessing
NVGaze [9]	1280×960	male 1-4 female 1-4	male 5 female 5	All	15623, 3895	↓4
OpenEDS[19] [38]	400×640	OpenEDS[19] train	OpenEDS[19] valid	PartSeg	8826, 2376	Crop to 400×300 ↑1.6
RITEyes General [47]	640×480	Avatars 1-18	Avatars 19-24	All	33997, 11519	↓2
LPW [35]	640×480	Subjects 1-16	Subjects 17-22	Pupil center	93127, 33388	↓2
Fuhl [4, 5]	384×288	I, III, VI, VIII, IX, XI, XIII, XV, XVII, XIX, XX, XXII	II, IV, V, VII, X, XII, XIV, XVI XVIII, XXI, XXIII	Pupil center	60079, 33846	↑5/3
PupilNet [50]	384×288	I, III, V	II, IV	Pupil center	25471, 15707	↑5/3

were pretrained on NVGaze, OpenEDS and RIT-Eyes training sets for two epochs.

5.2 Data augmentation

To increase the robustness of models and avoid overfitting, training images were randomly augmented with the following procedures with equal probability (12.5%) of occurrence:

- Horizontal flips
- Image rotation up to $\pm 30^{\circ}$
- Addition of Gaussian blur with $2 \leq \sigma \leq 7$
- Random Gamma correction for $\gamma = [0.6, 0.8, 1.2, 1.4]$
- Exposure offset up to ± 25 levels
- Gaussian noise with $2 \leq \sigma \leq 16$
- Image corruption by masking out pixels along a four-pixel thick line
- No augmentation

5.3 Evaluation Metrics

All segmentation performance is evaluated by IoU scores. Ellipse center accuracy is reported as the Euclidean distance in pixel error from their respective groundtruth annotations. Additionally, pupil and iris detection rate [11], *i.e.*, the percentage of ellipse centers accurately identified within a range of pixels of the groundtruth center point is also reported.

As most gaze estimation algorithms rely on ellipse fitting on the segmented pupil and/or iris, we quantify elliptical goodness of fit with metrics that effectively capture ellipse offset, orientation errors and scaling errors. In this work, we utilize a bounding box overlap IoU metric that accounts for all ellipse parameters: center, axes, and orientation. For each defined elliptical structure, a enclosing bounding box is generated. IoU scores are obtained from a comparison between groundtruth and predicted bounding boxes (Figure 6). Note that the orientation error (difference in ellipse orientation) of the fits is calculated for images in which the ratio of major to minor axis length exceeded 1.1 - this avoids large artifacts when elliptical fits are nearly circular.

6 Results and Discussion

6.1 Comparison with state-of-the-art models

The DenseElNet architecture is a hybrid of RITnet and TiramisuNet, and has 2.18M parameters. We also explore the alternative possibility of utilizing other state-of-the-art encoder-decoder architectures like DeepVOG and RITNet. DeepVOG, with 3.71M parameters, segments images into two classes; *pupil* and *background*, *i.e.*, (non-pupil). RITnet, with 0.25M parameters, defines four classes; *pupil*, *iris*, *sclera*, and *background* (other). Table 2 highlights that both RITnet and DenseElNet models outperform DeepVOG on every dataset. Table 2 also demonstrates that the performance of DenseElNet and RITnet are comparable ($< 2\%$ difference) on all datasets despite varying model complexity.

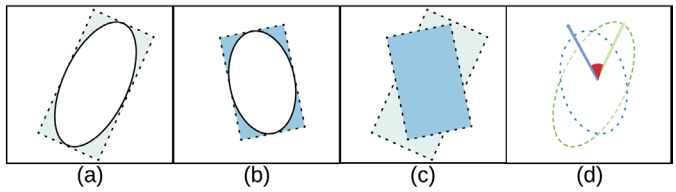

Fig. 6. Visualization of goodness of fit metrics used in the paper. (**a**) Groudtruth ellipse (pupil or iris). (**b**) Corresponding predicted ellipse. The rectangular boxes denote ellipse-axis-aligned bounding boxes for the respective ellipses. (**c**) denotes the bounding box overlap region and (**d**) illustrates the angular difference between the two ellipses.

Table 2. Eye Parts Segmentation: Comparison of *pupil* (and *iris*, inside parenthesis) *class* IoU scores for RITnet, DeepVOG and DenseElNet model architectures (along rows) in OpenEDS, NVGaze and RIT-Eyes dataset (along columns). Bold values indicate the best performance within each dataset. Because DeepVOG was not trained to segment the iris, we are unable to provide iris IOU scores.

Model	OpenEDS	NVGaze	RIT-Eyes
RITnet	95.0 (91.4)	**93.2 (91.7)**	89.5/94.4
DeepVOG	89.1 (NA)	90.9 (NA)	83.5 (NA)
DenseElNet	**95.4 (92.1)**	93.1 (91.4)	**91.5 (95.4)**

6.2 Ellipse center estimation

In this section, we explore the usefulness of the full ellipse segmentation (EllSeg) over the traditional eye parts segmentation (PartSeg) by comparing the pupil/iris center detection rates. We train three network architectures; RITnet, DeepVOG, and DenseElNet both with $\mathscr{L}_{SEG}$ loss functions using the following training scenarios:

- Traditional, four class PartSeg (referred as *RITnet-PartSeg*, *DeepVOG-PartSeg*, and *DenseElNet-PartSeg*)
- 3-class EllSeg (referred as *RITnet-EllSeg*, *DeepVOG-EllSeg*, and *DenseElNet-EllSeg*)

Table 3. The percentage of images classified as three categories of occlusion (see Section 6.2) for each dataset. Values are presented as pupil (iris).

	Occluded	Partial	Visible
OpenEDS	0.0 (0.0)	1.5 (17.2)	98.5 (82.7)
NVGaze	2.3 (0.0)	14.8 (75.6)	82.9 (24.4)
RITEyes	9.5 (11.1)	70.7 (22.3)	19.8 (66.7)

Note that, in this section, all ellipse centers are derived by utilizing ElliFit [51] along with RANSAC outlier removal on output segmentation maps.

Figure 7 presents the pupil/iris detection rate as a function of the error threshold (in pixels) for DeepVOG, RITnet, and DenseElNet, using both PartSeg and EllSeg frameworks. Although all models demonstrate similar performance when tested upon the OpenEDS dataset, models trained using the EllSeg framework demonstrate superior pupil and iris detection on the NVGaze and RIT-Eyes datasets.

Analysis of the ground truth imagery suggests that this difference may be attributed to the varying amounts of pupil/iris occlusion within each dataset. In order to verify this, we compute *occlusion magnitude*, O_m, which is defined as one minus the IoU of PartSeg and EllSeg ground truth maps. Based on this magnitude, each image is classified into 3 categories of occlusion (shown in Table 3) based on empirical thresholds, a) fully occluded ($O_m \geq 0.7$) b) partially occluded ($0.3 \leq O_m < 0.7$) and c) fully visible ($O_m < 0.3$).

Dramatic improvements can be observed for the NVGaze and RITEyes datasets wherein a large percent of images demonstrate partially occluded iris or pupil. Since a smaller percent of images are occluded in the OpenEDS dataset, we observe a small but consistent improvement in the iris detection rate between 3-6 pixel error threshold (see Figure 7, second row-first column). These results and subsequent analysis clearly demonstrate that EllSeg is robust to occlusions.

In addition to improving ellipse center estimates, Table 4 demonstrates that the EllSeg protocol reduces the number of images with invalid ellipse fits on the predicted segmentation output.

Table 4. The number of images without valid PartSeg or EllSeg ellipse fits for pupil (and iris, inside parenthesis) for DeepVOG, RITnet, and DenseElNet. The total column represents the number of valid images used for testing (as in section 4.1). Bold text (lower number) shows superior performance and illustrates the effectiveness of the EllSeg framework.

	Dataset	Total	DeepVOG	RITnet	DenseElNet
PartSeg	OpenEDS	2376	17 (NA)	1 (0)	2 (0)
	NVGaze	3895	10 (NA)	0 (0)	0 (0)
	RIT-Eyes	11519	1072 (NA)	287 (69)	353 (62)
EllSeg	OpenEDS	2376	**6** (NA)	1 (0)	**0** (0)
	NVGaze	3895	**0** (NA)	0 (0)	0 (0)
	RIT-Eyes	11519	**215** (NA)	**60** (**18**)	**1** (**0**)

6.3 Improving the ellipse estimates

In this section, we analyze the impact of L_{COM} on segmentation output maps, ellipse shape parameters and ellipse center estimates.

Ellipse center estimates results are shown in Figure 8. All models (RITnet, DeepVOG and DenseElNet) are trained with the EllSeg framework *with* and *without* $\mathscr{L}_{COM}$. Ellipse centers *without* $\mathscr{L}_{COM}$ loss are estimated using ElliFit on segmentation output maps. Models trained *with* $\mathscr{L}_{COM}$ loss estimate their centers (x_c and y_c) as shown in Figure 2.

Figure 8 also includes the results of non-CNN based algorithms ExCuSe [4], PuRe [8], and PuReST [7] which rely on filtered edges, morphological operations and handcrafted features using computer-vision based methods. Note that none of these methods were designed for OpenEDS, NVGaze, or RITeyes datasets. To facilitate application, pixels with a ground truth label identifying them as a member of the "background" class are converted to a uniform grey (digital count=127). This step minimizes the chance of false detection of the pupil within the background, which is a common issue for images within the OpenEDS and NVGaze datasets, which have black regions in the periphery. Note that for ExCuSe, images are resized to the author-recommended size (384x288). The predicted center is then remapped to (320x240) to facilitate comparison. For PuRe and PuReST, the EyeRecTool [55] is used to compute pupil center using the original image size (320x240).

Table 5. Comparison of Pupil center estimate errors (in pixels) on various datasets in terms of median scores. Note all the CNN models are trained with EllSeg framework. Image size is 320 × 240.

Model	RITnet		DenseElNet	
Method	Ellipse fit	$\mathscr{L}_{COM}$	Ellipse fit	$\mathscr{L}_{COM}$
OpenEDS	0.8	1.5	0.8	0.7
NVGaze	0.5	0.8	0.4	0.3
RIT-Eyes	1.0	1.2	0.7	0.7
Fuhl	-	73.4	-	1.7
LPW	-	4.7	-	0.8
PupilNet	-	77.6	-	1.6

Figure 8 reveals that, although introduction of $\mathscr{L}_{COM}$ often degraded the performance of RITnet, it improved performance for our model, (DenseElNet). Further, for pupil detection, the models trained using CNN outperforms all the non-CNNs based models ExCuSe, PuRe and PuReST.

Table 5 shows the comparison of median values of pupil center estimates *with* and *without* $\mathscr{L}_{COM}$ loss in regards to both models RITnet and DenseElNet. There is a slight improvement in the median values in the DenseElNet model with the introduction of this loss function. However, for the RITnet model, the inclusion of $\mathscr{L}_{COM}$ deteriorated the performance by 57%, 19%, and 19% for OpenEDS, NVGaze, and RIT-Eyes datasets respectively (within one-pixel error range for Pupil center). We suspect this behavior is due to the relatively limited channel size and low parameter count of RITnet when compared to DenseElNet.

The analyses presented up to this point focus on the accuracy of pupil/iris center estimates. However, many algorithms for gaze estimation rely on accurate estimation of pupil and iris ellipses for the construction of 3D geometric models of the oriented eye [12,14,15,31]. This necessitates a quantitative measure for the goodness of an ellipse fit. The methodology presented in Section 5.3 and represented in Figure 6 is used to calculate the *boundary IOU* - a measure used to estimate the quality of boundary estimation. Boundary IoU was calculated for both the pupil and the iris after application of RITnet and Densenet to several datasets, either with or without $\mathscr{L}_{COM}$. When $\mathscr{L}_{COM}$ is used, ellipse orientation and axis parameters are regressed via the bottleneck layer, and when it is not, the ellipse is fit to the segmented mask.

The result of this analysis are presented in Figure 9, and reveal that that DenseElNet *with* $\mathscr{L}_{COM}$ outperforms *without* $\mathscr{L}_{COM}$ in terms of boundary IOU and orientation error for both, the pupil and iris, on almost all datasets.

The pixel-wise IOU score of iris and pupil segmentation is presented in Figure 9 (last three rows). This analysis reveals that DenseElNet also outperforms other models in the segmentation of the pupil and iris. Although DeepVOG has the highest overall IoU score, one must also consider that the DeepVOG model is a two-class (binary) classifier (pupil vs. background) being compared against models of three-class segmentation (pupil, iris, background) and, in the former case, the IoU score is inflated by the presence of a large number or background pixels. This analysis also demonstrates that segmentation performance is improved by the inclusion of $\mathscr{L}_{COM}$ for all cases. Some examples of segmentation outputs with the inclusion of $\mathscr{L}_{COM}$ for OpenEDS and RIT-Eyes datasets are shown in Figure 10.

6.3.1 Qualitative Analysis: Effectiveness of $\mathscr{L}_{COM}$ loss

Here, we study the impact of the $\mathscr{L}_{COM}$ loss function with the DenseElNet architecture. Figure 11 shows the activation maps generated (*with*

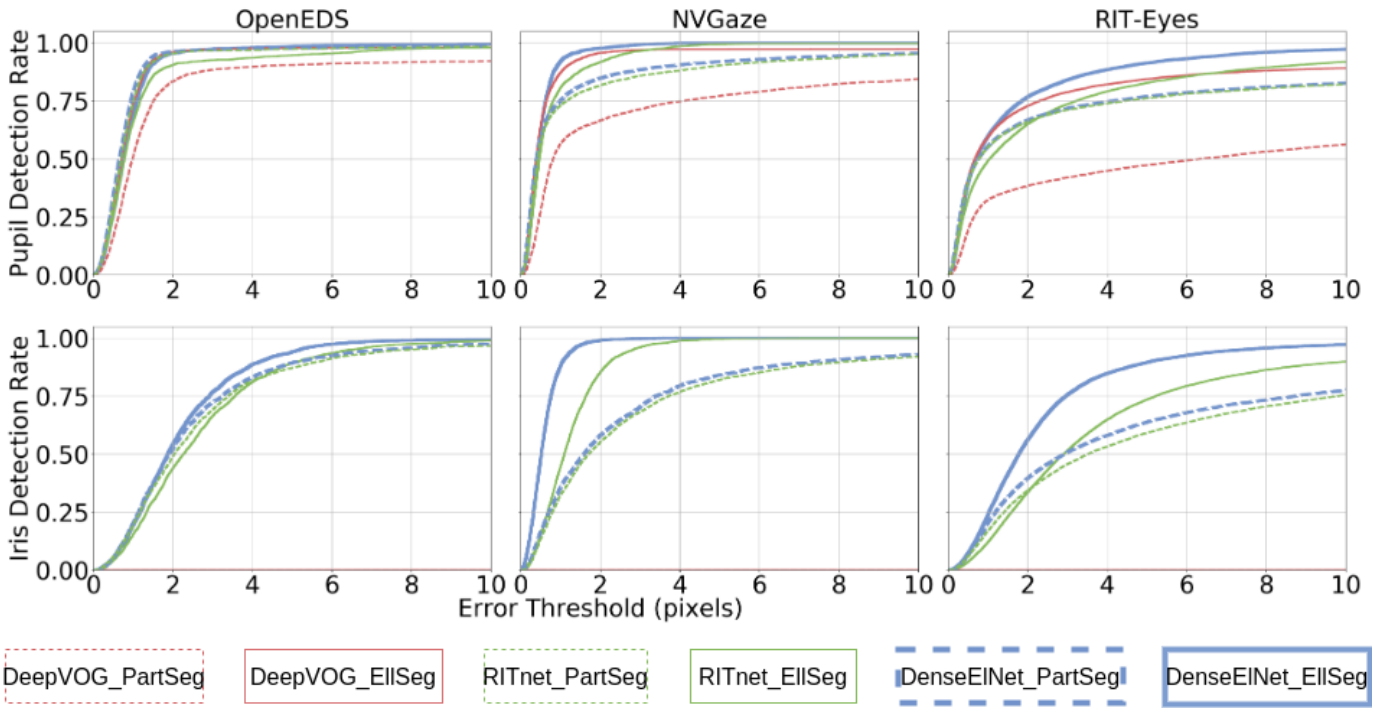

Fig. 7. *PartSeg vs EllSeg*: The pupil detection rate (top row) and iris detection rate (bottom row) as a function of the threshold for tolerated pixel error for center approximation for OpenEDS (left column), NVGaze (middle column) and RIT-Eyes (right column). Results for three architectures RITnet, DeepVOG and DenseElNet are present for both cases PartSeg (dashed lines) and EllSeg (solid lines). Note that only the pupil detection rate is shown for the DeepVOG architecture. All detection rates presented here are derived using ellipse fits on segmentation outputs on images sized at 320×240. Here, one pixel error corresponds to 0.25% of the image diagonal length.

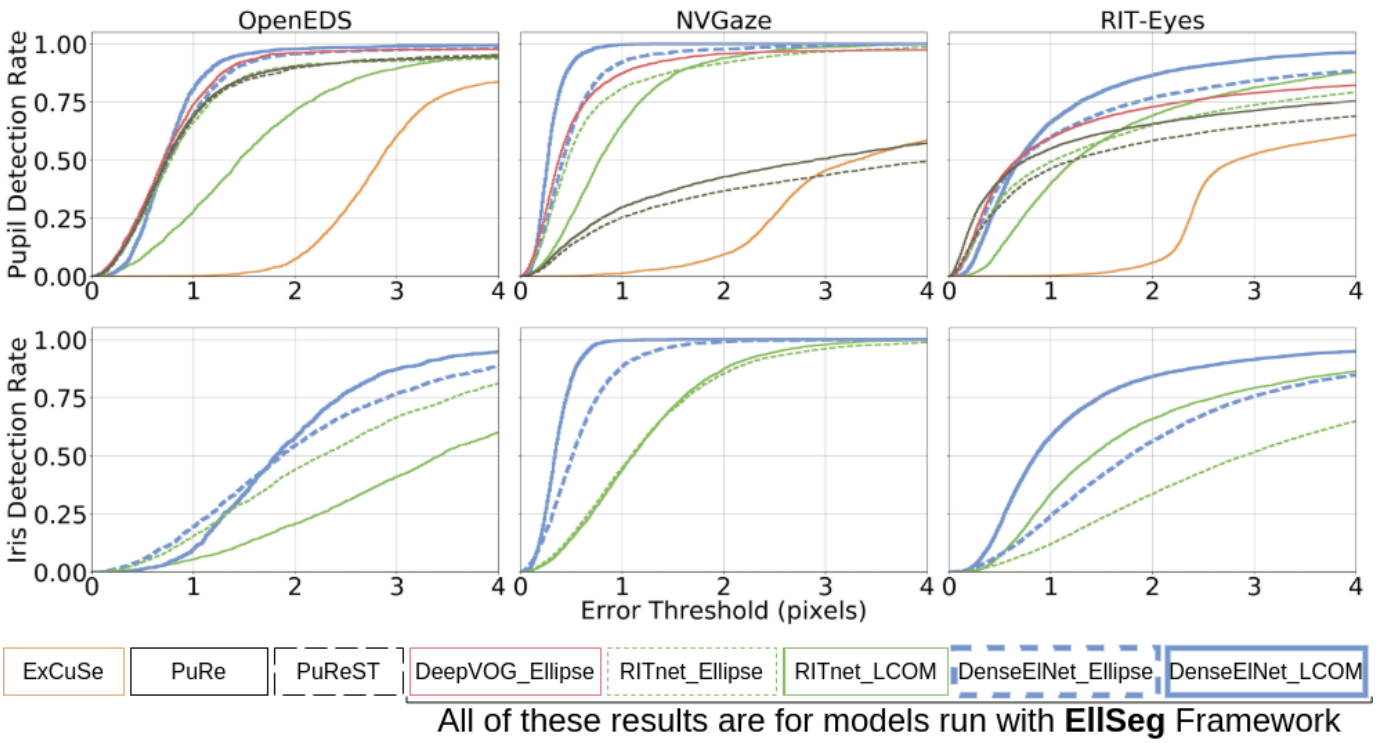

Fig. 8. *EllSeg with and without $\mathscr{L}_{COM}$ loss*: The pupil detection rate (top row) and iris detection rate (bottom row) for various pixel error thresholds of center approximation for three datasets. Models (RITnet, DenseElNet and DeepVOG) are trained with the EllSeg framework before the pupil center is estimated using either the ElliFit segmentation output map, or with $\mathscr{L}_{COM}$ loss. The result for non-CNN based model ExCuSe, PuRe and PuReST are also shown. One pixel error corresponds to 0.25% of the image diagonal length.

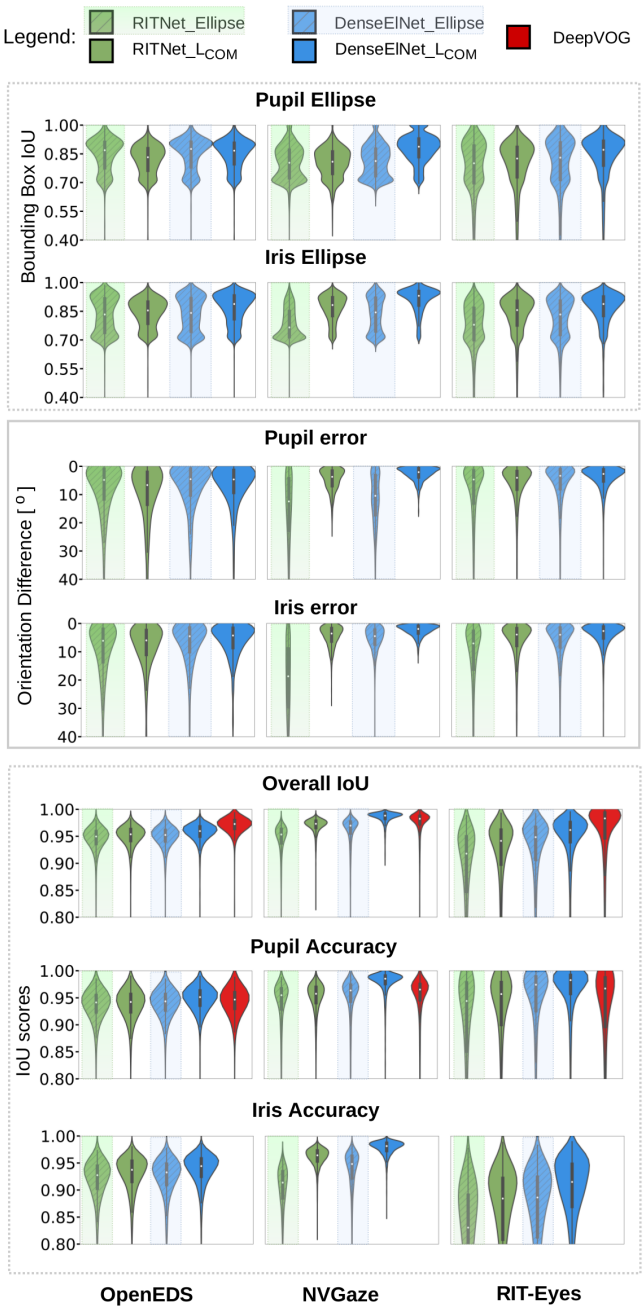

Fig. 9. Violin plots of boundary overlap IoU (1st and 2nd row: top dashed box), orientation error (3rd and 4th row: middle solid box), and segmentation IoU score (last three rows: bottom dashed box) following EllSeg framework by RITnet and DenseElNet, with or without $\mathscr{L}_{COM}$ loss ($\mathscr{L}_{COM}$ vs Ellipse), following application to the OpenEDS, NVGaze, and RIT-Eyes datasets (columns)
(Best viewed on screen).

and *without*) $\mathscr{L}_{COM}$ for three eye images. On closer observation of the pupil class, we observe a high intensity peak in the region around pupil center in the *with* $\mathscr{L}_{COM}$ condition (last column) compared to the *without* $\mathscr{L}_{COM}$ condition (fourth column from left). This peak around the pupil center is also evident in Figure 12 which shows a horizontal scan through the pupil center of one of the eye images illustrating the relative activation value for background, pupil, and iris *without* (left) and *with* (right) $\mathscr{L}_{COM}$.

Note that in Figure 11, the iris activation maps appear even when the iris is occluded by the eyelids in both *with* $\mathscr{L}_{COM}$ (second column from right) and *without* $\mathscr{L}_{COM}$ (third column from left) conditions.

Figure 12 shows relatively flat activation values near the iris centers for the iris class in both *with* and *without* $\mathscr{L}_{COM}$ cases; no peak is evident in the iris activation values. Note that the minimum in the background activation value localizes the center of the *iris* representing the inverse of the background (non-iris) region.

6.4 Center via bottleneck vs softargmax

To help provide an intuition regarding future network designs, we observe the impact of regressing the pupil and iris center estimates from the bottleneck (latent) layer [43], as opposed to estimating them using soft-argmax on the output segmentation maps (see Figure 2). Estimates from segmentation outputs are observed to be better than those regressed from latent space (pupil 81% $\rightarrow$ 98% and iris 42% $\rightarrow$ 58% detection at the two-pixel error margin) (see Figure 13). We hope that this intuition can help guide future efforts for CNN based near-eye feature extraction.

7 Summary

This paper presents EllSeg, a new framework for training a CNN to directly segment the entire elliptical structures of the pupil and iris. This framework was applied to RITnet [25], DeepVOG [14] and a custom designed hybrid model, DenseElNet, for segmentation as well as predicting pupil/iris ellipse estimates from eye images.

In Section 6.1, we benchmark our custom designed network architecture, DenseElNet, and achieve better baseline PartSeg performance to state-of-the-art encoder-decoder architectures, RITnet and DeepVOG (see Table 2). Our un-optimized forward pass implementation of DenseElNet operates at atleast 120Hz on a NVIDIA 1080 Ti, Intel-7800K. In Section 6.2, we show that our proposed framework *EllSeg* outperforms part-segmentation networks, *i.e.*, *PartSeg*, for pupil (OpenEDS: 0.2%, NVGaze: 11%, RIT-Eyes: 12%) and iris center (OpenEDS: 4%, NVGaze: 29%, RIT-Eyes: 25%) detection across three test datasets. Additional analysis reveals that the accuracy of EllSeg can be attributed to greater robustness to occlusion of the iris and pupil by the eyelids.

Section 6.3 demonstrates that the addition of $\mathscr{L}_{COM}$ loss function to the EllSeg framework results in improved pupil/iris ellipse estimates for pupil (OpenEDS: 2%, NVGaze: 11%, RIT-Eyes: 21%) and iris center (OpenEDS: 15%, NVGaze: 29%, RIT-Eyes: 40%) detection rate within a two-pixel error margin) and segmentation performance ($> 0.6\%, > 1.5\%, > 2\%$ for OpenEDS, NVGaze and RIT-Eyes respectively).

Visual inspection of output EllSeg activation maps reveals high confidence conditioned around the pupil and iris centers. Lastly in Section 6.4, we determine that deriving pupil and iris centers using softargmax is better than regressing the same via the bottleneck layer.

8 Conclusion and Future Work

To conclude, we present EllSeg, a simple 3-class full ellipse segmentation framework intended to extend conventional encoder-decoder architectures for the segmentation of eye images into pixels that represent the pupil, iris, and background. The EllSeg framework was benchmarked on multiple datasets using two network architectures: RITnet and our custom CNN design, DenseElNet. Results demonstrate superior estimation of the pupil and iris centers and orientation compared to their eye part segmentation models. An added benefit of the EllSeg framework is that it extends model training to image datasets in which only the pupil center has been labelled. Superior performance

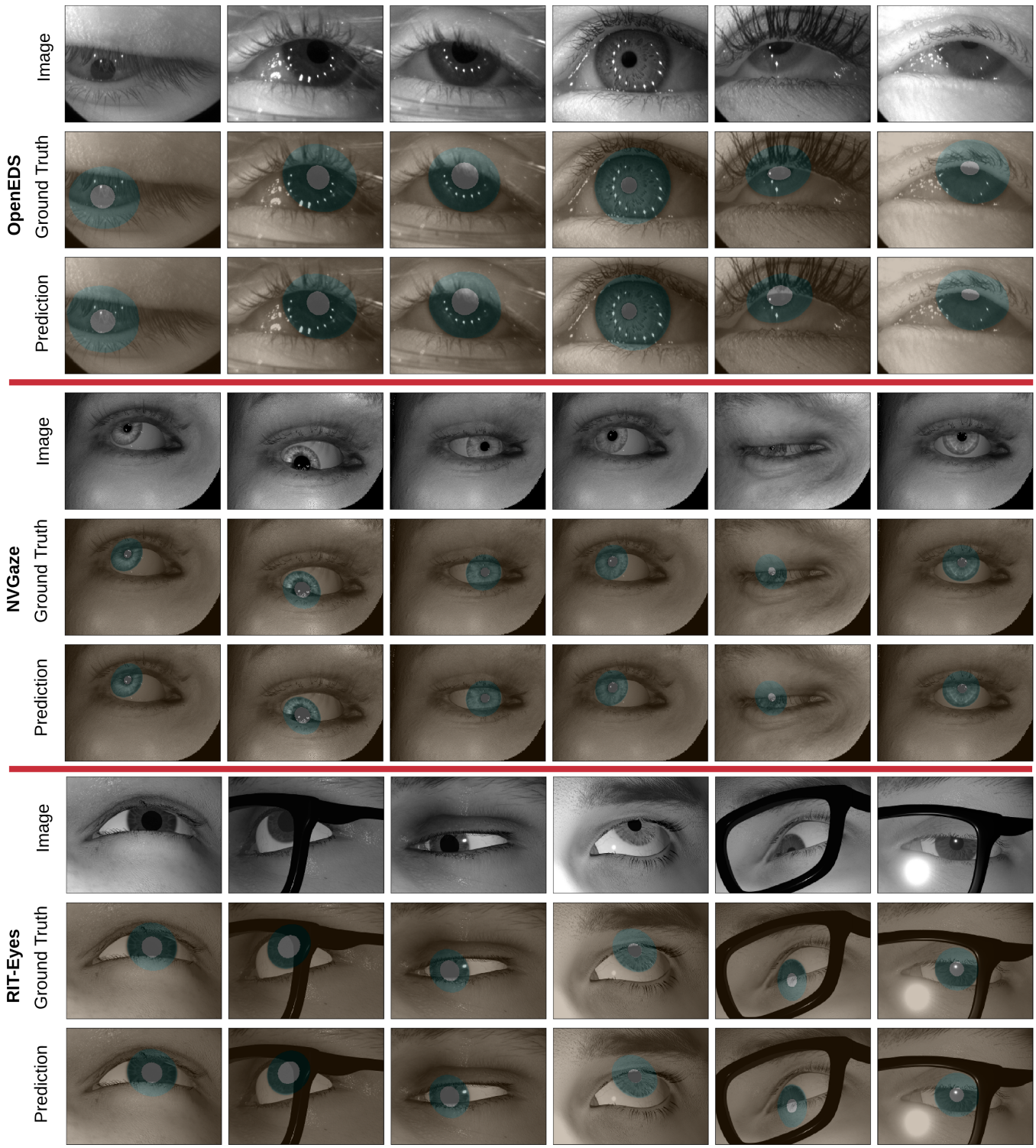

Fig. 10. DenseElNet model prediction and its respective ground truth for OpenEDS, NVGaze and RIT-Eyes dataset.

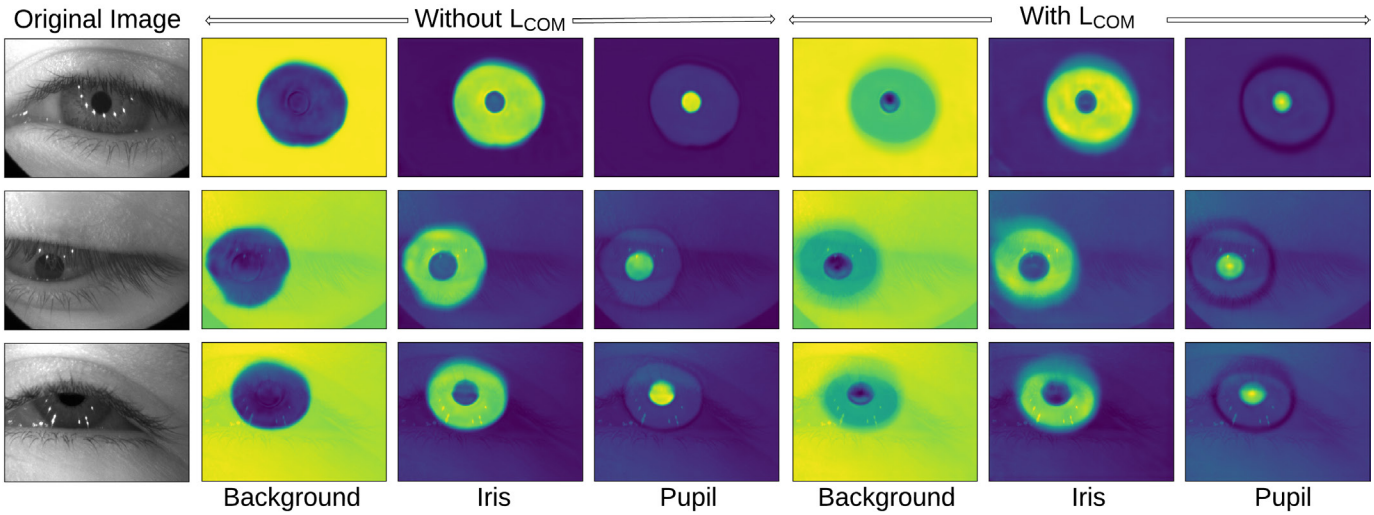

Fig. 11. Figure showing 2D activation maps. Columns (L-R): Original image (1st column), activation maps for background, iris and pupil class for model DenseElNet *without* $\mathscr{L}_{COM}$ (2nd-4th column) *with* $\mathscr{L}_{COM}$ (5th-7th column). Three rows show three different cases with bottom two having the original image in the background for reference. *(Best viewed on screen)*

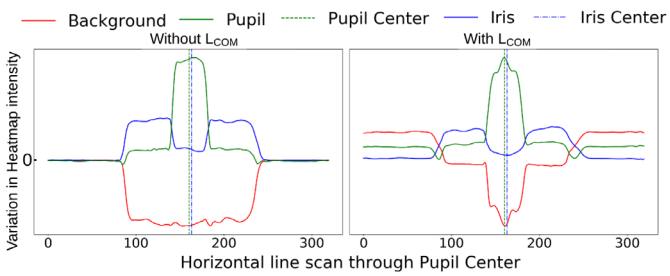

Fig. 12. A horizontal line scan across the pupil center to visualize DenseElNet output behavior without $\mathscr{L}_{COM}$ (left) and with $\mathscr{L}_{COM}$ (right). The inclusion of $\mathscr{L}_{COM}$ generates characteristic peaks which do not impede the task of semantic segmentation while effectively scaling output pixel activations near the predicted pupil and iris centers *(Best viewed on screen)*.

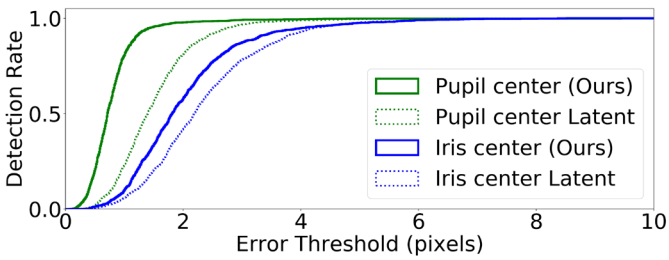

Fig. 13. The difference between pupil and iris detection rate in the OpenEDS dataset. Estimates are derived from the latent space and final segmentation maps (DenseElNet).

by the EllSeg framework can be attributed to greater robustness to occlusion of the pupil or iris.

While we evaluate EllSeg on multiple datasets collected from a large pool of individuals (see Table 1), a user based evaluation was not performed due to the time consuming nature of manual data collection and labelling. For future work, we intend on performing a comprehensive user study of our model on a wide range of subjects to further quantify the performance of our framework. We also intend on exploring other models with varying complexity to evaluate the efficacy of EllSeg. Pretrained models, code and other related resources will be made publicly available [1].

9 Acknowledgements

We thank Research Computing [56] at the Rochester Institute of Technology for providing all necessary hardware required for this project. We also thank Dr. Christopher Kanan for his helpful feedback and guidance. Lastly, we would like to thank Dr. Thiago Santini and Dr. Wolfgang Fuhl for their guidance in setting up the framework for their PuRe and PuReST algorithms.

References

[1] Steven A Cholewiak, Gordon D Love, Pratul P Srinivasan, Ren Ng, and Martin S Banks. Chromablur: Rendering chromatic eye aberration improves accommodation and realism. *ACM Transactions on Graphics (TOG)*, 36(6):1–12, 2017.

[2] Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. Towards foveated rendering for gaze-tracked virtual reality. *ACM Transactions on Graphics (TOG)*, 35(6):179, 2016.

[3] Andrew Duchowski. *Eye tracking methodology: Theory and practice*. 2007.

[4] Wolfgang Fuhl, Thomas Kübler, Katrin Sippel, Wolfgang Rosenstiel, and Enkelejda Kasneci. Excuse: Robust pupil detection in real-world scenarios. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9256:39–51, 2015.

[5] Wolfgang Fuhl, Thiago C. Santini, Thomas Kübler, and Enkelejda Kasneci. ElSe: Ellipse selection for robust pupil detection in real-world environments. In *Eye Tracking Research and Applications Symposium (ETRA)*, volume 14, pages 123–130, 2016.

[1] https://cis.rit.edu/~rsk3900/EllSeg/

[6] 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. 2017.

[7] Thiago Santini, Wolfgang Fuhl, and Enkelejda Kasneci. PuReST: Robust pupil tracking for real-time pervasive eye tracking. *Eye Tracking Research and Applications Symposium (ETRA)*, 2018.

[8] Thiago Santini, Wolfgang Fuhl, and Enkelejda Kasneci. PuRe: Robust pupil detection for real-time pervasive eye tracking. *Computer Vision and Image Understanding*, 170(February):40–50, 2018.

[9] Joohwan Kim, Michael Stengel, Alexander Majercik, Shalini De Mello, David Dunn, Samuli Laine, Morgan McGuire, and David Luebke. NVGaze: An anatomically-informed dataset for low-latency, near-eye gaze estimation. *Conference on Human Factors in Computing Systems - Proceedings*, 12:1–12, 2019.

[10] Wolfgang Fuhl, Thiago Santini, and Enkelejda Kasneci. Fast & robust eyelid outline & aperture detection in real-world scenarios. In *Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017*, pages 1089–1097. IEEE, 2017.

[11] Lech Świrski, Andreas Bulling, and Neil Dodgson. Robust real-time pupil tracking in highly off-axis images. *Eye Tracking Research and Applications Symposium (ETRA)*, pages 173–176, 2012.

[12] Lech Świrski and Neil A. Dodgson. A fully-automatic, temporal approach to single camera, glint-free 3D eye model fitting. *Pervasive Eye Tracking and Mobile Eye-Based Interaction (PETMEI)*, 2013.

[13] Jianfeng Li, Shigang Li, Tong Chen, and Yiguang Liu. A Geometry-Appearance-Based Pupil Detection Method for Near-Infrared Head-Mounted Cameras. *IEEE Access*, 6:23242–23252, 2018.

[14] 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*, 324:108307, 2019.

[15] Erroll Wood and Andreas Bulling. EyeTab: Model-based gaze estimation on unmodified tablet computers. *Eye Tracking Research and Applications Symposium (ETRA)*, pages 207–210, 2014.

[16] Haiyuan Wu, Qian Chen, and Toshikazu Wada. Conic-based algorithm for visual line estimation from one image, 2004.

[17] Alexander Plopski, Christian Nitschke, Kiyoshi Kiyokawa, Dieter Schmalstieg, and Haruo Takemura. Hybrid Eye Tracking: Combining Iris Contour and Corneal Imaging, 2015.

[18] Aayush Chaudhary and Jeff Pelz. Motion tracking of iris features to detect small eye movements. *Journal of Eye Movement Research*, 12(6):1–18, 2019.

[19] James K.Y. Ong and Thomas Haslwanter. Measuring torsional eye movements by tracking stable iris features. *Journal of Neuroscience Methods*, 2010.

[20] Yuta Itoh and Gudrun Klinker. Interaction-free calibration for optical see-through head-mounted displays based on 3d eye localization. In *2014 IEEE symposium on 3d user interfaces (3dui)*, pages 75–82. IEEE, 2014.

[21] Kamran Binaee, Gabriel Diaz, Jeff Pelz, and Flip Phillips. Binocular eye tracking calibration during a virtual ball catching task using head mounted display. In *Proceedings of the acm symposium on applied perception*, pages 15–18, 2016.

[22] Marcus Nyström, Richard Andersson, Kenneth Holmqvist, and Joost Van De Weijer. The influence of calibration method and eye physiology on eyetracking data quality. *Behavior research methods*, 45(1):272–288, 2013.

[23] Wolfgang Fuhl, Marc Tonsen, Andreas Bulling, and Enkelejda Kasneci. Pupil detection for head-mounted eye tracking in the wild: an evaluation of the state of the art. *Machine Vision and Applications*, 27(8):1275–1288, 2016.

[24] 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.

[25] 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 (ICCVW)*, pages 3698–3702. IEEE, 2019.

[26] Seonwook Park, Adrian Spurr, and Otmar Hilliges. Deep pictorial gaze estimation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11217 LNCS, pages 741–757. 2018.

[27] Susan M. Kolakowski and Jeff B. Pelz. Compensating for eye tracker camera movement. *Eye Tracking Research and Applications Symposium (ETRA)*, 2005(March):79–85, 2005.

[28] Thiago Santini, Diederick C. Niehorster, and Enkelejda Kasneci. Get a grip: Slippage-robust and glint-free gaze estimation for real-time pervasive head-mounted eye tracking. *Eye Tracking Research and Applications Symposium (ETRA)*, 2019.

[29] Pascal Bérard, Derek Bradley, Markus Gross, and Thabo Beeler. Lightweight eye capture using a parametric model. In *ACM Transactions on Graphics*, volume 35, 2016.

[30] David A. Atchison and Larry N. Thibos. Optical models of the human eye. *Clinical and Experimental Optometry*, 99(2):99–106, 2016.

[31] Moritz Kassner, William Patera, and Andreas Bulling. Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction. *UbiComp 2014 - Adjunct Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 1151–1160, 2014.

[32] Wolfgang Fuhl, Hong Gao, and Enkelejda Kasneci. Neural networks for optical vector and eye ball parameter estimation. In *ETRA Short Papers*, pages 4–1, 2020.

[33] Wolfgang Fuhl, Hong Gao, and Enkelejda Kasneci. Tiny convolution, decision tree, and binary neuronal networks for robust and real time pupil outline estimation. In *ETRA Short Papers*, pages 5–1, 2020.

[34] Zhengyang Wu, Srivignesh Rajendran, Tarrence van As, Joelle Zimmermann, Vijay Badrinarayanan, and Andrew Rabinovich. EyeNet: A Multi-Task Network for Off-Axis Eye Gaze Estimation and User Understanding. 2019.

[35] Marc Tonsen, Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Labelled pupils in the wild: A dataset for studying pupil detection in unconstrained environments. *Eye Tracking Research and Applications Symposium (ETRA)*, 14:139–142, 2016.

[36] Simon Jegou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2017-July:1175–1183, 2017.

[37] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning, 2016.

[38] Stephan J. Garbin, Yiru Shen, Immo Schuetz, Robert Cavin, Gregory Hughes, and Sachin S. Talathi. OpenEDS: Open Eye Dataset. 2019.

[39] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, 2015.

[40] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2016-Decem, pages 770–778, 2016.

[41] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9351:234–241, 2015.

[42] Reza Safaee-Rad, Ivo Tchoukanov, Kenneth Carless Smith, and Bensiyon Benhabib. Three-Dimensional Location Estimation of Circular Features for Machine Vision, 1992.

[43] Wolfgang Fuhl, Wolfgang Rosenstiel, and Enkelejda Kasneci. 500,000 Images Closer to Eyelid and Pupil Segmentation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11678 LNCS, pages 336–347. 2019.

[44] Aiden Nibali, Zhen He, Stuart Morgan, and Luke Prendergast. Numerical Coordinate Regression with Convolutional Neural Networks. 2018.

[45] E. Riba, D. Mishkin, D. Ponsa, and G. Bradski E. Rublee. Kornia: an open source differentiable computer vision library for pytorch. In *Winter Conference on Applications of Computer Vision*, 2020.

[46] Samuel Arba Mosquera, Shwetabh Verma, and Colm McAlinden. Centration axis in refractive surgery. *Eye and Vision*, 2(1):4, 2015.

[47] Nitinraj Nair, Aayush Kumar Chaudhary, Rakshit Sunil Kothari, Gabriel Jacob Diaz, Jeff B Pelz, and Reynold Bailey. Rit-eyes: realistically rendered eye images for eye-tracking applications. In *ACM Symposium on Eye Tracking Research and Applications*, pages 1–3, 2020.

[48] Carole H. Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss

function for highly unbalanced segmentations. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10553 LNCS:240–248, 2017.

[49] Hoel Kervadec, Jihene Bouchtiba, Christian Desrosiers, Eric Granger, Jose Dolz, and Ismail Ben Ayed. Boundary loss for highly unbalanced segmentation. pages 285–296, 12 2018.

[50] Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, and Enkelejda Kasneci. PupilNet: Convolutional Neural Networks for Robust Pupil Detection. 2016.

[51] Dilip K. Prasad, Maylor K.H. Leung, and Chai Quek. ElliFit: An unconstrained, non-iterative, least squares based geometric Ellipse Fitting method. *Pattern Recognition*, 46(5):1449–1465, 2013.

[52] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[53] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.

[54] Mark Everingham, Andrew Zisserman, Christopher KI Williams, Luc Van Gool, Moray Allan, Christopher M Bishop, Olivier Chapelle, Navneet Dalal, Thomas Deselaers, Gyuri Dorkö, et al. The 2005 pascal visual object classes challenge. In *Machine Learning Challenges Workshop*, pages 117–176. Springer, 2005.

[55] Thiago Santini, Wolfgang Fuhl, David Geisler, and Enkelejda Kasneci. Eyerectoo: Open-source software for real-time pervasive head-mounted eye tracking. In *VISIGRAPP (6: VISAPP)*, pages 96–101, 2017.

[56] Rochester Institute of Technology. Research computing services, 2019.