EyeGuide - From Gaze Data to Instance Segmentation

Jacqueline Kockwelp[1,2,3]
j.kockwelp@uni-muenster.de
Jörg Gromoll[2]
joerg.gromoll@ukmuenster.de
Joachim Wistuba[2]
joachim.wistuba@ukmuenster.de
Benjamin Risse[1,3]
b.risse@uni-muenster.de

[1] Institute for Geoinformatics
University of Münster
Münster, Germany

[2] Centre of Reproductive Medicine and Andrology
University Hospital Münster,
Münster, Germany

[3] Faculty of Mathematics and Computer Science
University of Münster,
Münster, Germany

Abstract

Obtaining precise instance-level segmentations is a challenging task in machine learning. Especially for objects with complex and non-convex geometries or with partial occlusions scribble, bounding boxes or user clicks are often provided to guide the segmentation. In this paper, we explore the usage of a remote eye tracking system to generate gaze data as an additional input for object segmentation models (called EyeGuide). The gaze data is recorded during routine image inspections (i.e. without giving a particular task) and is used as an additional input to train neural networks. Our results indicate that the acquisition of gaze data is faster and more convenient than providing explicit user input, less annotations are necessary to generate equal or better segmentation results and also overall better generalisation capabilities on unseen classes compared to state-of-the-art techniques. In summary, EyeGuide is a simple yet powerful guidance strategy that can directly be integrated in image inspection routines and neural network architectures.

1 Introduction

Supervised deep learning techniques have become the most powerful algorithms for a variety of computer vision tasks, including image classification, object detection and instance segmentation [6, 14]. This has been achieved by training these algorithms with huge amounts of usually manually annotated image or video data. As a consequence, a key challenge of developing these techniques has shifted from the development of sophisticated algorithms to providing abundant quantities of accurately curated training data, which is particularly true for image segmentation tasks requiring complex pixel-precise and dense annotations. For example, the recently published "segment anything model" has been developed to provide an efficient interactive data collection loop using a foundation model that has been

pretrained on 11 million images comprising 1 billion segmentation masks [11]. While foundation models such as SAM yield very good results on a variety of pre-defined image segmentation tasks user input is often still required to identify the particular object of interest. Moreover, given complex geometries, thin protrusions or partially occluded and disjoint object appearances, additional corrections are often provided in a semi-automatic and interactive fashion [26]. In addition many annotation tasks, such as segmenting particular cells in biomedical image data, require expert knowledge so that effective data acquisition is pivotal to enable precise image analyses at scale [3]. To avoid cumbersome and time consuming dense user input simplified prompts such as bounding boxes [5, 10, 13, 31], scribble [15, 25] or clicks [1, 12, 19, 23, 34] are often used to accelerate the data curation. Even though these types of inputs can be generated much faster than dense object contours, there is still a considerable amount of manual labour necessary to collect the annotations.

1.1 Contribution

In this paper we explore an alternative paradigm to provide an additional user generated input without the need for further explicit data interaction. We achieved this by integrating a remote eye tracking system into the regular data interaction workflow which captures the users gaze trajectories during the image inspection. Importantly, no specific tasks or constraints were given to the user so that the object was inspected freely for an arbitrary amount of time. We propose our method EyeGuide in which the resultant gaze data is converted into a gaze map and straight-forwardly integrated as an additional input to a neural network to guide the segmentation. To evaluate EyeGuide we generated gaze trajectories for PascalVOC2012 [7] and biomedical images from the Cellpose dataset [30] resulting in $4,007$ gaze annotations. Our results demonstrate that unconstrained gaze data is faster to acquire and often yields better results compared to other prompting techniques. Especially in case of complex highly non-convex geometries such as objects with frequent and thin protrusions or partially occluded objects, EyeGuide outperforms the state-of-the-art and results in more precise segmentation masks even when trained on a fraction of the training data. We therefore conclude that the integration of eye tracking data is a powerful and convenient alternative to other user input strategies and offers great potential for a variety of applications while reducing the needed labour time and improving the segmentation performance.

1.2 Related Work

Research on integrating gaze data into image analysis algorithms has been done for more than 10 years [20]. Our work is mainly related to two different approaches namely the integration of gaze data into supervised learning and instance segmentation with additional user input.

Gaze Data in Supervised Learning In the past gaze data has been integrated into supervised machine learning to extract attention heatmaps from human image inspections or to obtain auxiliary information for improving the machine learning tasks directly. To provide computer-aided diagnosis gaze data has been used to guide the attention of neural networks based on visual attention of a medical expert [32]. This has been achieved by integrating an attention consistency module which is used as a regularizer during training. Bhattacharya et al. also utilize the gaze data to improve the networks attention by suggesting a novel student-teacher architecture in combination with a visual attention loss [2]. Furthermore, Yu

et al. suggest to supervise neural attention models by human gaze in order to improve video captioning [36]. The authors introduce a gaze encoding network which provides spatial and temporal attention for sentence generations. Instead of generating attention maps from gaze, eye tracking data can also be used as auxiliary information directly. For example, Saab et al. used gaze data to improve medical image classifications and to estimate weak classification labels based on statistically derived gaze features [27]. Similarly, Wang et al. integrated gaze data to guide the region selection towards local information in a weakly supervised image classification framework [33]. Gaze trajectories have also been used as auxiliary information by converting it into gaze histograms, grid-based gaze features and sequential gaze-data which are subsequently used to enable zero-shot classification [9]. Beyond image classifications and to further simplify expert interactions Stember et al. integrated gaze data with speech recognition to add annotated 2D points of brain lesion locations in MRI scans [29]. The resultant dataset was used to train a neural network to predict single point locations with associated classes in a proof of principle study. Gaze data has also been used as auxiliary information to improve the learning of semantic visual attributes such as "open" or "pointy" [21]. To the best of our knowledge there exists only one work on eye tracking for deep learning-based segmentation that uses the gaze to generate segmentation masks which are subsequently used for weakly supervised learning [28]. Since the annotator was instructed to trace the objects contour a convex hull algorithm was sufficient to extract the masks directly from the eye movement.

Instance Segmentation with Additional User-Inputs In recent years additional inputs, such as bounding boxes and scribble, has been frequently used to improve performances. For example bounding boxes are used as noisy labels to improve boundary refinement [13] or as an additional input to provide soft constraints for a convolutional encoder-decoder network [35]. Scribble annotations, which provide dense point trajectories, are often used to improve semantic segmentation. For example one could use a scribbled initial contour surrounding the object which is then deformed to match the real boundary using a neural network [25]. In contrast to these two methods, individually set sparse clicks are one of the most commonly used additional inputs for instance segmentation. For example, Bearman et al. incorporated point supervision in combination with a novel objectness potential to a neural network [1]. Koohbanani uses either single clicks or scribbles to enable interactive segmentation for microscopy images [12]. Others have incorporated single clicks in the center of an imaginary bounding boxes into multiple instance learning techniques for weakly supervised object localization [23]. Alternatively, the same authors propose to provide four points on the object (top-, bottom-, left- and right-most point) followed by a GrabCut like segmentation algorithm [24]. The usage of extreme points for instance segmentation has been further explored by Maninis et al. [19]. In an approach called Deep Extreme Cut (DEXTR), the authors demonstrate that the segmentation accuracy strongly benefits from this additional input.

2 Method

An overview of our method is given in Figure 1. Using a remote eye tracking system, gaze trajectories are recorded and stored (see Figure 1a). Importantly, the gaze data is gathered indirectly, meaning that no particular task is given to the user (such as inspecting the object's boundary) and no additional explicit user input is required (such as drawing a bounding

box). Since the overall aim of our study is to explore the usage of gaze data as an additional input we evaluated different processing and gaze-specific augmentation and regularisation techniques. In a pre-processing step, the raw gaze information is filtered and aggregated (see Figure 1b) and the resulting points are merged to a map. Subsequently this map is concatenated with the RGB image to form a 4-channel input for the neural network (see Figure 1c). In contrast to click-based annotations, eye tracking results in densely sampled, yet noisy, gaze trajectories. To address these differences we studied the impact of blurred gaze maps, "gaze data jitter" and "gaze point dropout" on the segmentation performance and also evaluated the different performances using raw gaze data or fixation points.

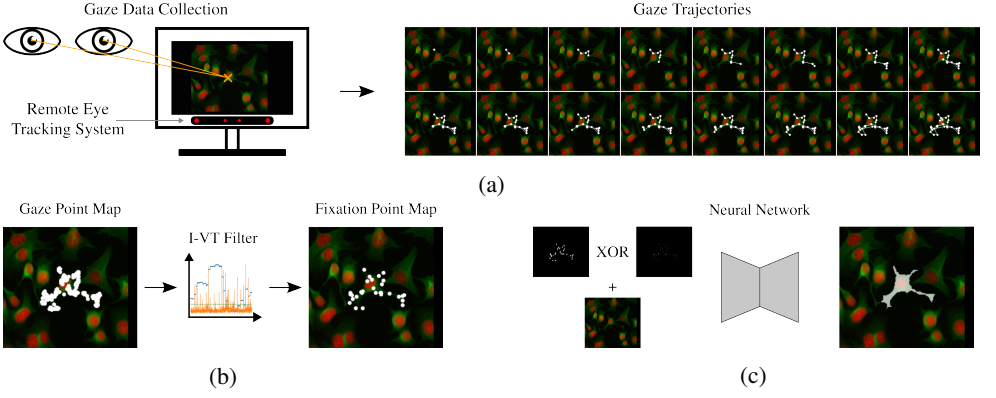


Figure 1: EyeGuide overview. a) Gaze information of a human annotator is recorded for individual object instances. b) Raw gaze information is filtered using an I-VT filter to generate fixation points. c) The raw gaze or fixation point map is concatenated with the RGB image and serves as the input for a neural network to predict instance segmentations.

Data Recording For the annotation and generation of our dataset, we implemented and used a simple annotation tool. We used the Tobii Pro Fusion eye tracking system, which is a screen-based remote eye tracker at 120 Hz.

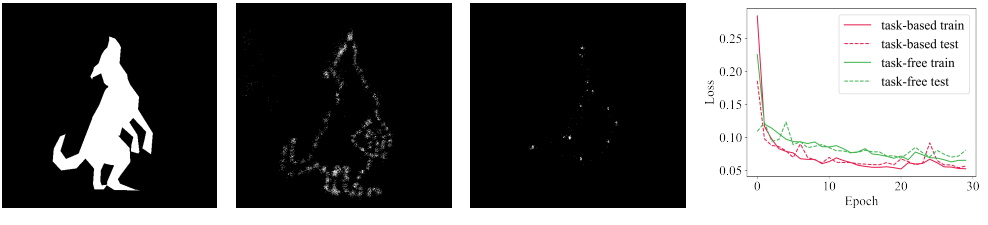
Preliminary Study To study the impact of gaze annotations on instance segmentation, we first analyzed different image inspection strategies, namely task-based versus task-free object inspection. For the task-based object inspection the annotator was asked to trace the boundary of an object, whereas for task-free inspection the user was only asked to look at the whole object of interest. In total, 564 objects of the COCO dataset [16] of the class dog were annotated by six human annotators (5 self-identified as male, 1 as female, mean age $\mu_{CP} = 25.83$, $\sigma_{CP} = 1.47$, 1 wore glasses) using the above mentioned strategies.

Task-free Annotation Dataset Generation Based on the results of our preliminary study, task-free object inspection results in faster and more comfortable user interactions while not compromising the segmentation performance of the neural network (see section 3). Therefore, the task-free inspection paradigm was used to generate our training dataset based on the following strategy: To inform the user about the object of interest the bounding box and the ground truth polygon mask were presented for 0.5 seconds. For very small objects initial zooms were applied and no time-limit was given for the observation. For evaluations and

processing, the user was asked to signal the object inspection start and end by pressing a button. Our tool also provides the possibility to repeat the observation and to move freely in the presented image by using standard mouse interactions (dragging, scrolling and zooming). This strategy was used to annotate PascalVOC2012 (train) and an additional biomedical dataset, namely a subset of the Cellpose dataset. PascalVOC2012 (train) contains $1,645$ images with $3,507$ individual instances, which are divided into 20 classes. These 20 classes can be grouped into four super-classes (person, animal, vehicle and indoor), while the number of images per class is nearly evenly distributed. The Cellpose dataset consists of a total of 608 labeled images, which show multiple types of cells and were acquired using various imaging methods. For our purposes we annotated 500 cell instances of the same type to also include highly non-convex objects as often found in biomedical applications. In total $4,007$ object instances were annotated by three different human annotators (2 self-identified as male, 1 as female, mean age $\mu_{PVOC} = 26.33$, $\sigma_{PVOC} = 2.08$, 1 wore glasses).

Gaze and Fixation Point Map Generation A raw gaze point consists of a (x,y) coordinate per eye and a timestamp. For gaze point map generation the coordinates of the right and left eye are combined into a map with a resolution equal to the input image. During fixations eyes essentially stop scanning and keep central foveal vision in place, enabling the visual system to perceive detailed information. In eye tracking data, fixation points are a union of multiple raw gaze points and consist of a (x,y) position and a start and end timestamp (i.e. duration). The most common technique to extract fixation points from gaze data is by using the I-VT filter [22]. This filter removes saccades (rapid eye fovea movements) and noise by thresholding the velocity of consecutive gaze points and aggregating the resultant locations into sparse fixations.

Gaze Data Augmentation To counteract overfitting and to study the impact of our weak and noisy annotations, we implemented two straight-forward augmentation methods for the gaze input. For the first method, called "gaze data jitter", normally distributed noise with $\mu = 0$ and $\sigma = 1.5$ in the range of -5 px to 5 px is added to the x- and y-coordinate of each raw gaze point. As a second variant "gaze data dropout" was implemented, where a percentage of the gaze points is randomly dropped.



(a) (b) (c) (d)

Figure 2: Comparison between task-based (contour tracing) and task-free image inspection. (a) Ground truth mask. (b) Gaze point map with explicit task. (c) Gaze point map with task-free inspection. (d) Training and test loss for task-based vs. task-free inspection.

Neural Network Architectures We test our algorithm using two different neural network backbones (ResNet-50, ResNet-101 [8]) and segmentation architectures (Fully-Convolutional

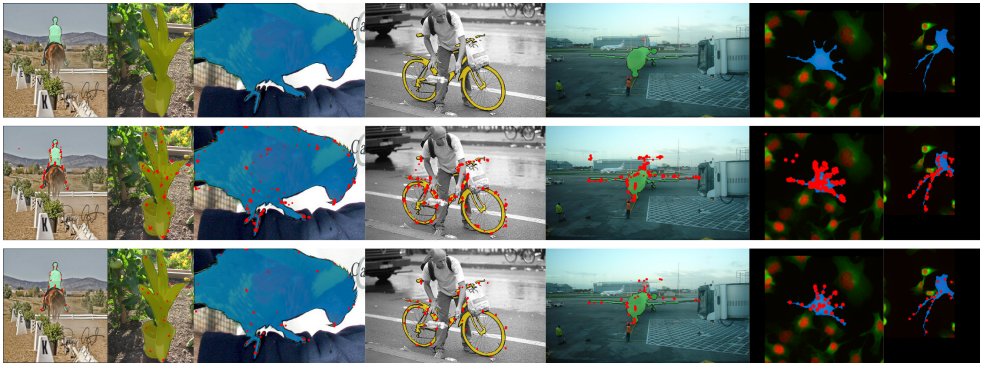
Network (FCN) [17] and DeepLabv3 [4]). All experiments for EyeGuide (with the exception of the preliminary study) were trained for 75 epochs each with a batch size of eight on a Nvidia RTX3090. AdamW [18] was used as the optimizer with the default parameter values and Binary Cross Entropy as the loss function. For evaluation purposes we compare our algorithm with DEXTR using the parameters as described by Maninis et al. [19].

3 Results

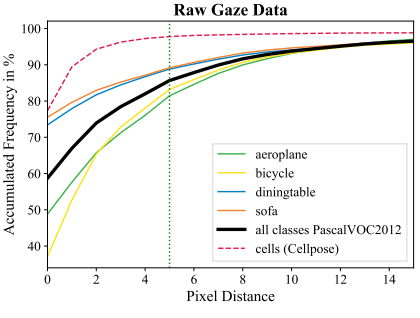
Preliminary Study We first evaluated the two visual object inspection strategies with respect to overall data acquisition efficiency, user-experience and segmentation performance. A comparison of the two resulting gaze trajectory visualisations between the contour tracing (see Figure 2b) and the task-free strategy (see Figure 2c) shows on the one hand that for the former significantly more data must be sampled and stored and on the other hand that this method takes more than twice as much time (13.3 seconds vs. 5.5 seconds). Furthermore all six annotators reported a more convenient user-experience in the task-free inspection, which was perceived as more natural and less exhausting and tiring. We next evaluated the segmentation performance when both inspection strategies are used as an additional input for instance segmentation. As can be seen in Figure 2d the difference between the two object inspection tasks is negligible. Based on these preliminary evaluations we therefore focused on the task-free annotation strategy in all our following experiments.

Annotation Efficiency The average annotation time was 6.19 seconds for PascalVOC2012 and 5.13 seconds for Cellpose. For comparison, drawing a polygon mask takes an average of 79 seconds, adding a bounding box takes 35 seconds and clicking extreme points takes 7.5 seconds respectively [19, 24]. Figure 3a shows some examples of our generated datasets together with visualizations for the gaze and fixation point maps. From the gaze point maps (second row) it can be seen that the entire object has been scanned. It should be noted that even parts of objects were considered which were not included in the ground truth mask but belong to the object of interest (Figure 3a, column four, tail of the aeroplane). Additionally, with the help of the I-VT filter the noise level is reduced so that the focus and localisation is limited to the object (third row).

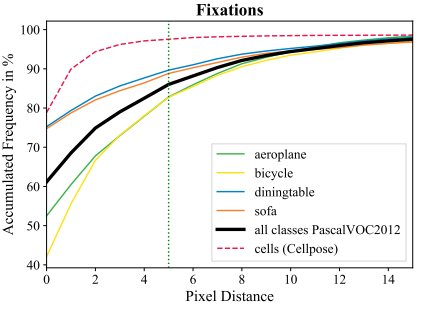
Gaze Accuracy A certain inaccuracy cannot be excluded when using eye tracking, despite user specific calibration, which can also be caused by e.g. the distraction of the annotator. We analysed our datasets by calculating the accumulated frequency in percentage of raw gaze data points (see Figure 3b) and fixations (see Figure 3c) grouped by their distance to the ground truth mask. Here one can see that the differences between the two variants are only marginal. It can be seen that the majority of the collected gaze information lies within or very close to the ground truth mask (black line; mean of all PascalVOC2012 classes). Only for objects with complex geometries and fine structures, such as bicycles, fixations scatter strongly around the object while still being located very close to the ground truth mask (e.g. bicycle class). For comparison we included the "undefined region" threshold of PascalVOC2012 as a vertical dashed line in Figure 4b, which is defined by a 5 pixel boundary around the contour reflecting the inter-user variability (called "void pixel" [7]).



(a)



(b)



(c)

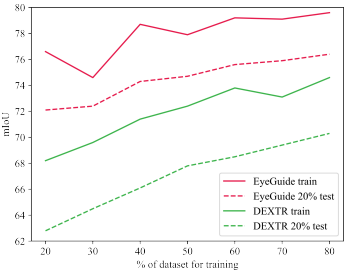
Figure 3: (a) Dataset examples for PascalVOC2012 and Cellpose. **First row:** Images with ground truth mask. **Second row:** Images with gaze point map visualization. **Third row:** Images with fixation point map visualization. (b) Raw Gaze Data (c) Fixations grouped by their distance to the ground truth mask of PascalVOC2012 (mean of all classes and four selected classes) and Cellpose dataset.

Instance Segmentation Performance For the development of EyeGuide we trained and evaluated different neural network architectures and model configurations. An overview of these experiments is listed in Table 4a. For the initial tests we used a Fully-Convolutional Network (FCN) with ResNet-50 as backbone. We explicitly excluded pretraining in our experiments to investigate the performance of EyeGuide without any prior knowledge. Moreover and in contrast to several other instance segmentation evaluations (e.g. [19]), we did not exclude the "undefined region" pixel at the object boundaries to better evaluate the performance for non-convex and thin objects, resulting in a more conservative error estimation.

In our first experiment we used the concatenated RGB image and gaze point map without any further processing as the input for the neural network. Since the object of interest can be localised on the basis of the filtered fixation points, automatically extracted bounding boxes (with an optional margin; we used 50 pixel in our experiments) can be used to crop the instances. Cropping improved the mean Intersection over Union (mIoU) by 22% so all further experiments were performed with cropping. Next we evaluated the impact of oversampling by using the fixation point map instead of the gaze map as an additional input. A decrease in performance could be observed indicating that the neural network benefits from

Architecture	fixation filter	blurred	gaze data jitter	gaze data dropout	mIoU PascalVOC2012	mIoU Cellpose
FCN+ResNet-50 (without cropping)					53.2	-
FCN+ResNet-50					75.4	82.9
FCN+ResNet-50	x				74.6	82.7
FCN+ResNet-50		x			75.8	83.3
FCN+ResNet-50			x		75.9	81.8
FCN+ResNet-50			x	x (30%)	75.7	-
FCN+ResNet-50				x (30%)	76.3	83.4
FCN+ResNet-101					76.2	-
DeepLabv3+ResNet-50					75.9	-
FCN+ResNet-101				x (30%)	**76.4**	-
FCN+ResNet-50		x		x (30%)	76.2	**83.6**
DEXTR					70.3	78.9
DEXTR (pretrained)					75.9	82.7

(a)



(b)

Figure 4: Experiments overview. a) Evaluation of different architectures and model configurations for PascalVOC2012 and Cellpose. b) Evaluation of the amount of training data on the performance and learning progress of EyeGuide vs. DEXTR.

the more densely sampled raw gaze data. Therefore, further experiments were done on the gaze point maps only. The impact of noise was studied by applying a 5×5 2D Gaussian kernel on the gaze data which improved the mIoU by 0.4% for both datasets. As expected, similar improvements could be achieved by using the above mentioned gaze data jitter, however, no substantial performance gains could be reached using this augmentation technique. Considering slightly more consistent performance improvements using gaze dropout (in our experiments best performances were achieved with a 30% dropout rate) we chose this augmentation technique to also counteract potential overfitting effects. As further architectures, we tested ResNet-101 as backbone for the FCN and DeepLabv3 with backbone ResNet-50. Our analyses revealed that the best performance was achieved using a FCN with ResNet-101.

Figure 5: Prediction examples for PascalVOC2012 and Cellpose. **First row:** Image with gaze points. **Second row:** Predicted mask. **Third row:** Ground truth mask.

Furthermore, we evaluated the influence of the amount of training data on the performance for EyeGuide and DEXTR (see Figure 4b). Our method achieves significantly better results than DEXTR with a considerably smaller amount of data. In comparison, DEXTR achieves a similar mIoU only with 80% of the training data, which EyeGuide accomplishes with 20% of the data. A qualitative evaluation of EyeGuide can be found in Figure 5. The first five images are taken from the PascalVOC2012 test dataset and were only used for testing. The fourth column shows an example of how our predicted masks are often even better

than the ground truth mask, since the human annotator omitted the saddle when looking at the horse, the neural network also excluded it for the segmentation mask prediction. This architecture was further used and compared with Deep Extreme Cuts (DEXTR), being the current state-of-the-art for weakly annotated instance segmentation in both, annotation efficiency and segmentation performance [19]. We trained and evaluated DEXTR on PascalVOC2012 and Cellpose with the same 20/80 train/test split. No pretraining was used for the ResNet backbone to allow better comparability between the methods. As can be seen in Table 4a EyeGuide outperforms DEXTR by +6.1% on the PascalVOC2012 and +4.4% on the Cellpose dataset. Interestingly, even when comparing our non-pretrained segmentation algorithm with DEXTR using a pretrained backbone our method still generates slightly better results (+0.5% on PascalVOC2012 and +0.6% on Cellpose). In another experiment, we evaluated the generalisability (see Table 1) of our methodology compared to DEXTR by training one network per super-class of PascalVOC2012 and evaluating on the unseen classes. The training dataset for the super-classes ranged from 680 to 752 images. On the test dataset of the seen classes, we reached a mIoU improvement between +5.1% and +11.9% compared to DEXTR, where EyeGuide is particularly superior in learning the super-class person. Also in the evaluation of the unseen super-classes, our method shows significantly better results for each test, averaging +10.3% for person, +3.3% for vehicle, +11.6% for animal and +5.1% for indoor.

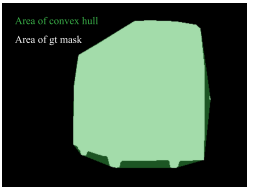
	seen classes						unseen classes											
	train			test			person			vehicle			animal			indoor		
	# img	mIoU Ours	mIoU DEXTR	# img	mIoU Ours	mIoU DEXTR	# img	mIoU Ours	mIoU DEXTR	# img	mIoU Ours	mIoU DEXTR	# img	mIoU Ours	mIoU DEXTR	# img	mIoU Ours	mIoU DEXTR
person_net	696	77.8	68.3	174	**75.1**	63.2	-	-	-	852	**65.6**	60.7	847	**78.2**	64.5	938	**65.2**	59.2
vehicle_net	685	76.5	70.4	167	**71.3**	64.8	870	**67.8**	56.8	-	-	-	847	**73.8**	64.1	938	**63.2**	58.4
animal_net	680	85.2	78.9	167	**81.1**	74.3	870	**71.3**	61.2	852	**64.1**	62.0	-	-	-	938	**63.7**	59.2
indoor_net	752	72.1	66.1	186	**66.6**	61.5	870	**69.0**	58.6	852	**63.2**	60.1	847	**73.9**	62.5	-	-	-
all_net	2,813	79.6	74.6	694	**76.4**	70.3	-	-	-	-	-	-	-	-	-	-	-	-

Table 1: Evaluation of the generalisation capabilities of EyeGuide vs. DEXTR to unseen classes. mIoU results for training on a subset of classes and testing on the left out subsets.

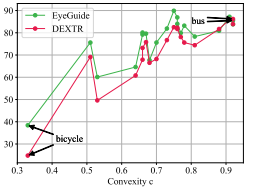
Segmentation Performance Objects with Low Convexity The qualitative and quantitative evaluations indicated a potential correlation between the shape of the objects and the EyeGuide performance. Since human annotators gaze over the entire object, additional gaze information is available for (almost) all regions. This observation is particularly interesting when comparing our annotation strategy with other user inputs. Especially given complex geometries with frequent and thin protrusions, providing (extreme) points, scribble or accurately placed bounding boxes can be a cumbersome and time consuming task. To further investigate this relationship between shape complexity and segmentation accuracy we computed the convexity for each class of PascalVOC2012, which we defined as $c = \frac{\text{area of gt mask}}{\text{convex hull area of gt mask}}$, $c \in [0,1]$.

The results are summarized in Figure 6. As can be seen, the convexity varies strongly between the different classes. It is noticeable that EyeGuide shows a huge performance improvement of up to +13.6% for all classes that have a convexity < 0.88. Especially for classes with very fine, irregular shapes (e.g. bicycle and chair) our method is superior. Only for very quadrangular structures, e.g., busses and trains, the mIoU of DEXTR is 0.1% to 0.8% higher than the mIoU of our method. It should be noted that a non-pretrained EyeGuide still outperforms a pretrained DEXTR on highly convex classes indicating that our approach can be beneficial in situations with no or only a few existing annotations (as is for example often the case in medical applications).

(a)



(b)



(c)

	bicycle	aeroplane	chair	potted plant	motorbike	horse	bird	dining table	person	cow	dog	cat	sheep	boat	sofa	car	bottle	tv/ monitor	bus	train	mIoU
Convexity c	0.33	0.52	0.52	0.64	0.66	0.66	0.67	0.68	0.70	0.73	0.76	0.76	0.77	0.77	0.78	0.81	0.88	0.91	0.92	0.92	
EyeGuide	**38.4**	**75.6**	**60.1**	**64.6**	**79.2**	**80.1**	**79.6**	**67.7**	**75.6**	**81.9**	**89.9**	**86.9**	**84.0**	**79.9**	**83.2**	**78.3**	80.9	**87.2**	85.5	83.8	**77.1**
DEXTR	24.8	69.1	49.6	60.8	67.9	73.2	75.8	66.5	68.2	76.7	82.5	82.3	81.7	78.1	75.6	74.4	**81.7**	85.9	**86.2**	**83.9**	72.2
DEXTR (pretrained)	32.6	80.0	55.6	62.8	76.8	79.3	81.5	67.0	74.4	79.4	89.7	86.5	86.9	82.7	82.5	79.8	82.4	88.7	88.9	86.6	77.2

(d)

Figure 6: Class-wise evaluation on PascalVOC2012 dataset. (a) Bicycle ground truth mask ($c = 0.25$). (b) Bus ground truth mask ($c = 0.95$). (c) Convexity mIoU plot for every class. (d) Detailed table including comparison with DEXTR (pretrained) for every class.

4 Conclusion

In this paper we propose a novel guiding strategy called EyeGuide which integrates remote eye tracking data as additional input for neural networks to guide the training of instance segmentation. Compared to other weakly guided methods our approach offers the advantage of generating these inputs indirectly during regular image inspections (e.g. in the medical field) without the need for additional and explicit user input. EyeGuide shows significantly better performance compared to the state-of-the-art, especially for objects that have fine or irregular geometries. Moreover, we show that less training data is required to achieve good segmentation results and the generalisability of the neural network model is significantly improved, making our methodology suitable for annotating new and complex datasets.

Data availability: The gaze data for the PascalVOC2012 train dataset and the 500 cells of Cellpose dataset can be found at https://doi.org/10.17879/29928498088.

Acknowledgements JK, JG, JW and BR would like to thank the Deutsche Forschungsgemeinschaft (DFG) - CRU326. BR would also like to thank the Deutsche Forschungsgemeinschaft (DFG) – CRC 1450 (431460824). The calculations for this work were performed on the computer cluster PALMA II of the University of Münster. The authors would like to thank Sebastian Thiele for the valuable comments and suggestions and Marlon Becker and Pascal Kockwelp for their feedback on the manuscript.

References

[1] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the Point: Semantic Segmentation with Point Supervision. *Lecture Notes in Computer Science*, pages 549–565, 2016. ISSN 0302-9743. doi: 10.1007/978-3-319-46478-7_34.

[2] Moinak Bhattacharya, Shubham Jain, and Prateek Prasanna. RadioTransformer: A Cascaded Global-Focal Transformer for Visual Attention–Guided Disease Classifica-

tion. *Lecture Notes in Computer Science*, pages 679–698, 2022. ISSN 0302-9743. doi: 10.1007/978-3-031-19803-8_40.

[3] Gabriel Chartrand, Phillip M Cheng, Eugene Vorontsov, Michal Drozdzal, Simon Turcotte, Christopher J Pal, Samuel Kadoury, and An Tang. Deep Learning: A Primer for Radiologists. *RadioGraphics*, 37(7):2113–2131, 2017. ISSN 0271-5333. doi: 10.1148/rg.2017170077. Reference for: 1 full-time month of work for two expert radiologists to label 1000 images.

[4] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. arxiv 2017. *arXiv preprint arXiv:1706.05587*, 2, 2019.

[5] Jifeng Dai, Kaiming He, and Jian Sun. BoxSup: Exploiting Bounding Boxes to Supervise Convolutional Networks for Semantic Segmentation. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1635–1643, 2015. doi: 10.1109/iccv.2015.191.

[6] Shi Dong, Ping Wang, and Khushnood Abbas. A survey on deep learning and its applications. *Computer Science Review*, 40:100379, 2021. ISSN 1574-0137. doi: 10.1016/j.cosrev.2021.100379.

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Nour Karessli, Zeynep Akata, Bernt Schiele, and Andreas Bulling. Gaze Embeddings for Zero-Shot Image Classification. *CVPR*, 2016. doi: 10.48550/arxiv.1611.09309.

[10] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple Does It: Weakly Supervised Instance and Semantic Segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1665–1674, 2017. doi: 10.1109/cvpr.2017.181.

[11] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[12] Navid Alemi Koohbanani, Mostafa Jahanifar, Neda Zamani Tajadin, and Nasir Rajpoot. NuClick: A Deep Learning Framework for Interactive Segmentation of Microscopy Images. *arXiv*, 2020. doi: 10.48550/arxiv.2005.14511.

[13] Viveka Kulharia, Siddhartha Chandra, Amit Agrawal, Philip Torr, and Ambrish Tyagi. Box2Seg: Attention Weighted Loss and Discriminative Feature Learning for Weakly Supervised Segmentation. *Lecture Notes in Computer Science*, pages 290–308, 2020. ISSN 0302-9743. doi: 10.1007/978-3-030-58583-9_18.

[14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553): 436 – 444, 05 2015. ISSN 0028-0836. doi: 10.1038/nature14539.

[15] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. ScribbleSup: Scribble-Supervised Convolutional Networks for Semantic Segmentation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3159–3167, 2016. doi: 10.1109/cvpr.2016.344.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[17] 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.

[18] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[19] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool. Deep extreme cut: From extreme points to object segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[20] Stefan Mathe and Cristian Sminchisescu. Actions in the eye: Dynamic gaze datasets and learnt saliency models for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(7):1408–1424, 2014.

[21] Nils Murrugarra-Llerena and Adriana Kovashka. Learning Attributes from Human Gaze. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 510–519, 2017. doi: 10.1109/wacv.2017.63.

[22] Anneli Olsen. The tobii ivt fixation filter algorithm description. 2012.

[23] Dim P Papadopoulos, Jasper R R Uijlings, Frank Keller, and Vittorio Ferrari. Training object class detectors with click supervision. *CVPR*, 2017. doi: 10.48550/arxiv.1704.06189.

[24] Dim P. Papadopoulos, Jasper R. R. Uijlings, Frank Keller, and Vittorio Ferrari. Extreme clicking for efficient object annotation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4940–4949, 2017. doi: 10.1109/ICCV.2017.528.

[25] Sida Peng, Wen Jiang, Huaijin Pi, Xiuli Li, Hujun Bao, and Xiaowei Zhou. Deep snake for real-time instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[26] Hiba Ramadan, Chaymae Lachqar, and Hamid Tairi. A survey of recent interactive image segmentation methods. *Computational Visual Media*, 6(4):355–384, 2020. ISSN 2096-0433. doi: 10.1007/s41095-020-0177-5.

[27] Khaled Saab, Sarah M. Hooper, Nimit S. Sohoni, Jupinder Parmar, Brian Pogatchnik, Sen Wu, Jared A. Dunnmon, Hongyang R. Zhang, Daniel Rubin, and Christopher Ré. Observational Supervision for Medical Image Classification Using Gaze Data. *Lecture Notes in Computer Science*, pages 603–614, 2021. ISSN 0302-9743. doi: 10.1007/978-3-030-87196-3_56.

[28] J N Stember, H Celik, E Krupinski, P D Chang, S Mutasa, B J Wood, A Lignelli, G Moonis, L H Schwartz, S Jambawalikar, and U Bagci. Eye tracking for deep learning segmentation using convolutional neural networks. *J Digit Imaging*, 32(4):597–604, August 2019.

[29] Joseph N. Stember, Haydar Celik, David Gutman, Nathaniel Swinburne, Robert Young, Sarah Eskreis-Winkler, Andrei Holodny, Sachin Jambawalikar, Bradford J. Wood, Peter D. Chang, Elizabeth Krupinski, and Ulas Bagci. Integrating Eye Tracking and Speech Recognition Accurately Annotates MR Brain Images for Deep Learning: Proof of Principle. *Radiology: Artificial Intelligence*, 3(1):e200047, 2020. doi: 10.1148/ryai.2020200047.

[30] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, 18(1):100–106, 2021.

[31] Zhi Tian, Chunhua Shen, Xinlong Wang, and Hao Chen. BoxInst: High-Performance Instance Segmentation with Box Annotations. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 00:5439–5448, 2021. doi: 10.1109/cvpr46437.2021.00540.

[32] Sheng Wang, Xi Ouyang, Tianming Liu, Qian Wang, Dinggang Shen, and Sheng Wang. Follow My Eye: Using Gaze to Supervise Computer-Aided Diagnosis. *IEEE Transactions on Medical Imaging*, 41(7):1688–1698, 2022. ISSN 0278-0062. doi: 10.1109/tmi.2022.3146973.

[33] Xin Wang, Nicolas Thome, and Matthieu Cord. Gaze latent support vector machine for image classification improved by weakly supervised region selection. *Pattern Recognition*, 72:59–71, 2017. ISSN 0031-3203. doi: 10.1016/j.patcog.2017.07.001.

[34] Ning Xu, Brian Price, Scott Cohen, Jimei Yang, and Thomas Huang. Deep Interactive Object Selection. *CVPR*, 2016. doi: 10.48550/arxiv.1603.04042.

[35] Ning Xu, Brian Price, Scott Cohen, Jimei Yang, and Thomas Huang. Deep GrabCut for Object Selection. *BMVC*, 2017. doi: 10.48550/arxiv.1707.00243.

[36] Youngjae Yu, Jongwook Choi, Yeonhwa Kim, Kyung Yoo, Sang-Hun Lee, and Gunhee Kim. Supervising Neural Attention Models for Video Captioning by Human Gaze Data. *CVPR*, 2017. doi: 10.48550/arxiv.1707.06029.