

# Panoptic Visual Analytics of Eye Tracking Data

Valeria Garro<sup>a</sup> and Veronica Sundstedt<sup>b</sup>

Blekinge Institute of Technology, Karlskrona, Sweden

**Keywords:** Eye Tracking, Visualization, Semantic Areas of Interest, Panoptic Segmentation.


**Abstract:** In eye tracking data visualization, areas of interest (AOIs) are widely adopted to analyze specific regions of the stimulus. We propose a visual analytics tool that leverages panoptic segmentation to automatically divide the whole image or frame video in semantic AOIs. A set of AOI-based visualization techniques are available to analyze the fixation data based on these semantic AOIs. Moreover, we propose a modified version of radial transition graph visualizations adapted to the extracted semantic AOIs and a new visualization technique also based on radial transition graphs. Two application examples illustrate the potential of this approach and are used to discuss its usefulness and limitations.


## 1 INTRODUCTION

Several visualization techniques have been proposed to perform analysis of eye tracking data providing information about the location of the user's visual attention and its variations on a stimulus. The use of Areas of Interest (AOIs) is a well-established method for eye tracking data analysis to study how participants' attention is distributed over particular regions (Blascheck et al., 2017a). AOIs are specific regions of the stimulus which are highly significant; they can have a specific meaning for the analyst, commonly a semantic meaning (e.g. a specific object in the scene), or they can be identified directly using the gaze data, e.g. clustering of fixations (Blascheck et al., 2017a). The definition of AOIs plays a crucial role in the analysis, and it is a fundamental part of the study's hypothesis (Holmqvist et al., 2015); missing or inaccurate AOIs can limit the results. A common practice in eye tracking analysis is to define AOIs by manual annotation. This process is time-consuming, especially for video stimuli (Holmqvist et al., 2015), and prone to spatial inaccuracies. In some cases, this manual process is necessary due to the need of the analyst to define a particular area of the stimuli. However, in the case of AOIs with *semantic meaning* (e.g. object classes like vehicles, people, and furniture), computer vision methods for image object detection and image classification can support the AOIs extraction and automate the process. Recently several

automatic AOIs extraction methods based on object detection have been proposed, due also to the latest improvement of performance and accuracy of deep learning approaches (Wolf et al., 2018) (Panetta et al., 2020) (Barz and Sonntag, 2021). These automatic approaches also support the analysis of eye tracking data gathered from head-mounted devices, e.g. eye tracking glasses. In this case, the recording sessions are usually long and they differ between participants due to the nature of egocentric footage; hence a manual definition of AOIs would be time-consuming.

Following the recent advancements in image segmentation research, in this position paper we investigate the use of *panoptic segmentation* (Kirillov et al., 2019b) to divide the entire stimulus into different areas with semantic meaning, from here on denoted as Semantic AOIs (SAOIs), and we apply several AOI-based visualization techniques to analyze the eye tracking data. In computer vision, image parsing (Tu et al., 2005) or scene parsing (Tighe et al., 2014), more recently addressed as panoptic segmentation (Kirillov et al., 2019b), can be defined as the combination of semantic segmentation (Shelhamer et al., 2017) and instance segmentation. With semantic segmentation, we address the problem of assigning a semantic class label to every pixel of the image (or video frame) with no distinction between different instances of the same class (e.g. two cars belong to the same class). Combining this to instance segmentation, panoptic segmentation also distinguishes between different instances of specific countable classes. We argue that this technique applied to the analysis of eye tracking data allows a

<sup>a</sup>  <https://orcid.org/0000-0002-9527-4594>

<sup>b</sup>  <https://orcid.org/0000-0003-3639-9327>

comprehensive semantic analysis of the entire stimulus, not limiting the analysis to a set of predefined AOIs. For this reason, we explore a novel approach of visual analytics of eye tracking data which leverages a holistic semantic scene parsing of the stimuli. More in detail: (i) we present a prototype of our visual analytics tool of eye tracking data which automatically extracts SAOIs from the stimulus using deep learning panoptic segmentation; (ii) the visual analytics tool includes a set of established AOI-based visualization techniques showing both temporal and relational features, among which we propose a variant of the AOI radial transition graph (Blascheck et al., 2013) (Blascheck et al., 2017b) adapted to the concept of SAOIs, and a novel AOI transition graph also based on the radial transition graph. Finally, two application examples are included to illustrate the potential of the tool performing a semantic AOIs analysis of eye tracking data and discussing its limitations.

## 2 RELATED WORK

AOI visual analytics of eye tracking data consists of two main steps: the AOIs definition and the selection of visualization techniques to analyze the data. In this section, we group relevant previous works based on these two phases.

### 2.1 AOI Definition

AOIs are usually created by manual annotation, e.g. defining the region of interest with simple shapes like rectangles or more complex polygon shapes. Alternatively, a common approach is the automatic generation of AOIs by processing the eye tracking data (Blascheck et al., 2017a), e.g. spatial clustering of fixation locations (Privitera and Stark, 2000) (Santella and DeCarlo, 2004), and processing fixation heatmaps (Fuhl et al., 2018a) in image stimuli, as well as space-time clustering of fixations in video stimuli (Kurzahls and Weiskopf, 2013). A third approach is based on processing the stimulus (image or video) and extracting AOIs based on saliency maps (Fuhl et al., 2018b), (Privitera and Stark, 2000), (Borji and Itti, 2013).

Recent works investigated the application of machine learning image segmentation algorithms supporting AOIs extraction, also in conjunction with the need to analyze eye tracking data gathered from head-mounted devices. In (Wolf et al., 2018), the authors proposed a method for automatic gaze mapping on AOIs extracted based on Mask R-CNN (He et al., 2017), a deep learning object detection and segmen-

tation algorithm. In their pilot study, they trained the algorithm with a relatively small dataset limiting the object detection to two different classes of objects in a controlled test environment. They compared the automatic gaze mapping to a manual mapping considered as ground truth. The evaluation showed promising results but also highlighted the impact of the size of the training dataset when applying a deep learning-based algorithm. Recently, Barz and Sonntag presented two methods for automatic AOIs detection based on pre-trained deep learning models (Barz and Sonntag, 2021). The first one classifies fixed-size image patches centered on the gaze signal using ResNet, while the second one has a similar approach of (Wolf et al., 2018) but uses a Mask R-CNN model pre-trained on the MS COCO dataset (Lin et al., 2014).

The use of image segmentation algorithms for AOIs extraction as part of visual analytics tools of eye tracking data has been proposed by (Panetta et al., 2020). The authors presented ISeeColor, a visualization system of eye tracking data that defines AOIs using deep learning-based image semantic segmentation algorithms, i.e. Deeplabv3 (Chen et al., 2018a) (Chen et al., 2018b) and EncNet (Zhang et al., 2018). The system is designed for the analysis of egocentric eye tracking data and it automatically annotates objects of interest (OOI) according to a predefined set of semantic categories, e.g. cars and people. The fixation data are analyzed with respect to the extracted OOI and visualized in the system. In addition to classic eye tracking data visualizations, e.g. scarf plot (Richardson and Dale, 2005), the fixation duration of each OOI is also visualized by a recoloring of the segmented OOI overlaid on the video frames. Different colors represent different fixation durations.

Our work differs from IseeColor as we investigate the use of panoptic segmentation, a holistic scene parsing of the entire image, instead of the extraction of only specific classes. We also provide different visualization techniques to analyze the fixation data; in particular, we add visualization techniques that analyze the relation between AOIs, such as AOI transition graphs. Moreover, we propose two new variants of the AOI radial transition graph.

### 2.2 AOI-based Visualization Techniques

According to (Blascheck et al., 2017a), AOI-based visualizations can be categorized in two main approaches: one drawing attention to AOIs temporal visualizations, and the other highlighting the relationship between AOIs.

Timeline AOI visualizations represent the gaze data in relation to the AOIs focusing on the tempo-

ral feature. A typical example of temporal visualization is the scarf plot (Richardson and Dale, 2005), a color-coded timeline representing the focus of the participant over time on the set of AOIs in which each AOI is represented by a different color. Scarf plots of several participants can be aligned and displayed close to each other in the same view for comparison. Parallel Scan-Path (Raschke et al., 2012) and AOI Rivers (Burch et al., 2013) are examples of timeline visualizations representing in one dimension the time while in the other dimension the set of predefined AOIs. While the scarf plot is unique for each participant, these visualizations can intrinsically display gaze data from multiple participants. In Parallel Scan-Path, the data of each participant are shown individually while in the AOI rivers in an aggregated form.

Alternatively, relational AOI visualizations focus on displaying the relation among AOIs, e.g. transitions between AOIs. AOI transitions can be visualized in different ways. A simple approach is an AOI transition matrix (Goldberg and Kotval, 1999) in which rows and columns correspond to the AOIs and the value at cell  $(i, j)$  represents the frequency of transitions from AOI  $i$  to AOI  $j$  of two consecutive fixations. Examples of more complex visualization techniques are AOI transition trees (Kurzahls and Weiskopf, 2015), and AOI circular transition diagrams (Blascheck et al., 2013) also called radial transition graphs (Blascheck et al., 2017b). In the radial transition graph, the AOIs are represented in a circular diagram as ring sectors. In the internal part of the circular diagram, lines connecting two sectors depict transitions between the two AOIs, while the thickness of the line encodes the transition frequency. Only the AOIs which were focused on by the participant are displayed in the layout. Several variants of this type of visualization have been proposed varying the size and the colors of the ring sectors. For example, the sector size can be equal for all AOIs or proportional to the aggregated fixation duration within an AOI, while the color can encode the fixation count or identify a specific AOI. In (Blascheck et al., 2017b), the authors presented a graph comparison method based on radial transition graphs. In their method, they applied a version of radial transition graph in which the size of the sectors is proportional to the aggregated fixation duration and the colors identify different AOIs. Each radial transition graph displays the eye tracking data of one participant. This circular and compact layout supports a direct comparison of a pair of participants based on the juxtaposition of their corresponding radial transition graphs (Blascheck et al., 2017b).

Our proposed visualization tool includes both temporal (scarf plot) and relational AOI-based visualizations. Our version of the radial transition graph, i.e. the *SAOI radial transition graph*, also encodes the area of the extracted AOIs. Moreover, we also propose a further modification called *SAOI mirror radial transition graph* in which the transition lines follow a predefined pattern that could improve the readability.

### 3 PANOPTIC VISUAL ANALYTICS TOOL

The proposed visualization tool is implemented in Python and all visualizations are created with the Matplotlib library (Hunter, 2007). An overview of the interface is shown in Figure 1a. The user loads the stimulus (image or video) via the interface and can choose between starting the segmentation algorithm or directly loading the segmentation data in case the segmentation has already been performed. The panoptic segmentation is performed using Detectron2 (Wu et al., 2019), Facebook AI Research library platform which provides state-of-the-art computer vision detection and segmentation algorithms, and it is based on PyTorch (Paszke et al., 2019). The panoptic segmentation algorithm available in Detectron2 is an implementation of the work of (Kirillov et al., 2019a) called Panoptic Feature Pyramid Networks (Panoptic FPN) based on Mask R-CNN. In the case of video stimuli, the current implementation of panoptic segmentation in Detectron2 also provides a basic propagation of instance IDs across frames which is suitable for scenes that do not present major overlaps between different instances.

The output of the segmentation consists of a segmented image or a set of frames in which the color of each pixel represents a specific semantic AOI, and a corresponding text file provides information about the association between colors and SAOIs. When the user loads the eye tracking data, the tool processes the fixation data and assigns each fixation to a SAOI according to the fixation location. This process is performed by analyzing the colors of a  $5 \times 5$  mask centered on the location of the fixation and extracting the SAOI corresponding to the most frequent color on the mask. The SAOIs extracted in the segmentation phase are displayed on the lower left part of the visualization tool. They are ordered according to the percentage of occupied area in the stimuli. This ordering allows an initial analysis of the SAOIs and facilitates the selection of the relevant SAOIs and the exclusion of those SAOIs which have been classified incorrectly by the segmentation algorithm. When the user imports the

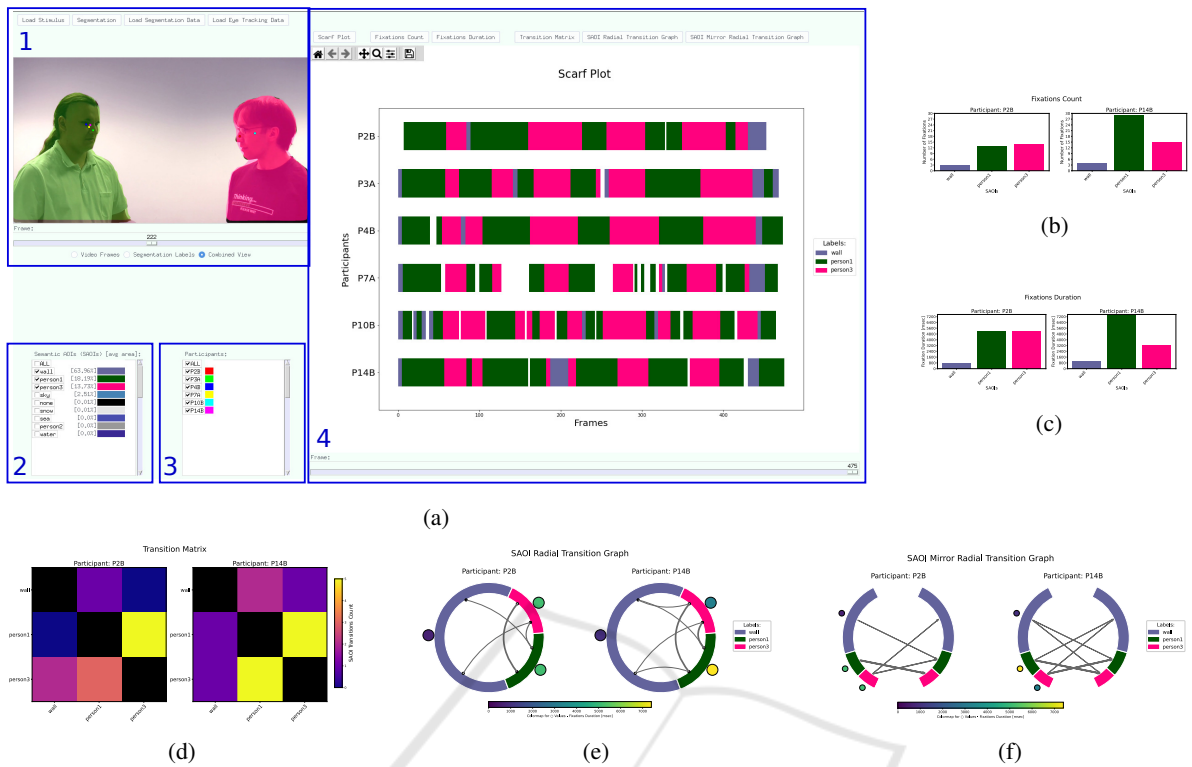


Figure 1: (a) Panoptic Visual Analytics Tool. (1) The stimuli view shows the stimuli frame and the segmentation results together with the fixation locations for the selected participants. (2) List of SAOIs extracted from the panoptic segmentation algorithm. (3) List of participants from the loaded eye tracking data. (4) Visualization view in which the user can choose between six different visualizations; here the scarf plot is shown from six participants. (b)-(f) The other available visualizations, in this example only for participants P2B and P14B and the selected SAOIs: (b) Fixation count bar graph, (c) Fixation duration bar graph, (d) Transition matrix, (e) SAOI radial transition graph, and (f) SAOI mirror radial transition.

eye tracking data on the tool, the list of participants appears on the right of the list of extracted SAOIs. The stimuli view on the top left can be used to inspect the semantic AOIs and also the fixation locations. The user can choose among the original stimuli, the color-coded segmented stimuli, and a combination of these two views by a superimposition of the semi-transparent version of the segmentation over the original stimuli. The fixation locations are represented by small colored circles with the corresponding color of the participant shown in the participants list.

A set of six different visualizations are available to analyze the fixation data of the selected participants over the selected SAOIs: a scarf plot, bar graphs representing fixation counts and fixation durations, a transition matrix, a modified version of the radial transition graph, and a novel transition diagram which we call *SAOI mirror radial transition graph*. For all visualizations, the analyst can navigate through time using the frame slider at the bottom of the visualization area. The selected visualization is then updated showing only the data related to the time span up to the frame indicated by the slider. The adapted version of

the radial transition graph, which we call *SAOI radial transition graph*, is presented in the following section, while the description of the components of the novel *SAOI mirror radial transition graph* is presented in Section 3.2.

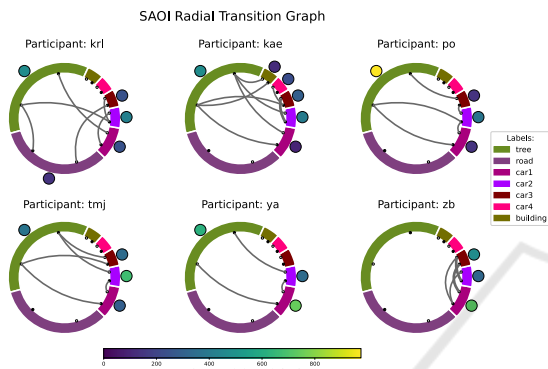
### 3.1 SAOI Radial Transition Graph

The SAOI radial transition graph is a modified version of the radial transition graph (Blascheck et al., 2017b) which has been described in Section 2.2. In our version, the size of a ring sector is proportional to the percentage of the area covered by the corresponding SAOI on the stimuli. Moreover, all SAOIs selected by the user are visualized in the graph, not exclusively the ones focused on by the participant. Each ring sector has the same color of its corresponding SAOI from the segmentation data to facilitate the data correlation between the stimuli view and the visualization view.

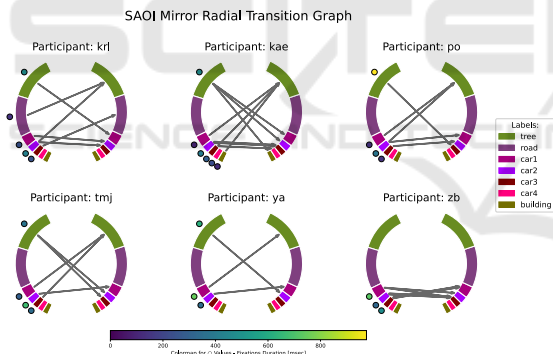
The fixation duration value of each SAOI is displayed with a color-coded circle positioned in the external part of the circular layout and it is centered to its corresponding ring sector, as shown in Figure 1e.



(a)



(b)



(c)

Figure 2: Panoptic Visual Analytics Tool extracting SAOIs with panoptic segmentation from an image of the FixaTons dataset. (a) Tool interface and scarf plot visualization. (b) SAOI radial transition graph. (c) SAOI mirror radial transition graph.

The applied colormap is a global colormap ranging from zero to the highest fixation duration extracted from all selected participants' data. Only the ring sectors of SAOIs focused on by the participant are complemented with the corresponding fixation duration circles. The transitions between SAOIs are encoded with a transition line in the internal part of the circular graph. The thickness of the line is proportional to the number of transitions. As in (Blascheck et al., 2017b),

each ring sector has two distinct anchor points for the transition lines to avoid visual clutter, one representing the starting point of a transition and the other the ending point.

### 3.2 SAOI Mirror Radial Transition Graph

The proposed novel visualization called *SAOI mirror radial transition graph* is based on our SAOI radial transition graph. The layout of the diagram is still circular but each SAOI is represented twice in the graph with two ring sectors positioned symmetrically with respect to the vertical axis, as shown in Figure 1f. This design has been inspired by the connectogram (Irimia et al., 2012), a graph representation of brain regions' connectivity, which has a symmetrical layout of the two cerebral hemispheres.

A transition between two SAOIs  $s_i$  and  $s_j$  is encoded with an arrow line starting from the ring sector on the left side of the graph corresponding to  $s_i$  and ending on the ring sector on the right side of the graph corresponding to  $s_j$ . The proposed layout would allow better readability of the transitions compared to the radial transition graph, especially for transitions between two adjacent SAOIs and in the case of a graph with numerous SAOIs. The starting SAOIs are always located on the left side of the diagram while the ending SAOIs are always on the right side. Moreover, it does not require two distinct anchor points in each ring sector. As in the SAOI radial transition graph, the size of each ring sector is proportional to the area of the corresponding SAOI and the thickness of the arrow lines corresponds to the number of transitions. The fixation duration of each SAOI is still represented by a color-coded circle positioned near the matching ring sector but only on the left side of the graph to avoid visual clutter.

## 4 APPLICATION EXAMPLES

As a first analysis, we show the capabilities of our visualization tool, and the two proposed visualizations are presented by two application examples, i.e. an image stimulus and a video stimulus. The image stimulus belongs to the FixaTons dataset (Zanca et al., 2018), a collection of datasets' scanpaths, i.e. ordered sequences of fixations, including stimuli from the MIT1003 dataset (Judd et al., 2009). The video stimulus is included in the Eye Tracking Benchmark dataset (Kurzahls et al., 2014). For both examples, we used the Panoptic FPN model pretrained on COCO (R101-FPN) available on the Detectron2 website (Wu

et al., 2019). In the first example, we test our visualization tool with an image from the FixaTons dataset showing an outdoor scene with four cars on a road. We run the panoptic segmentation on the image with a confidence threshold of 0.6, i.e. keeping instance predictions with a confidence score higher or equal to 0.6. The algorithm correctly extracted four instances of the class ‘car’, one instance of the class ‘stop sign’, and the following uncountable classes: ‘tree’, ‘road’, ‘sky’, and ‘building’, as shown in the stimuli view of the interface in Figure 2a. In Figure 2, we show three different visualizations available in our tool for the data of six participants. The scarf plot in Figure 2a reveals the timeline of the SAOIs focused on by each participant during the three seconds free-viewing sequence. From the scarf plot, it is possible to notice the difference in the scanpaths between participants. For example, the fixations of the last participant (‘zb’) focused more on the ‘car’ SAOIs, shifting attention between car instances, while participant ‘po’ looked longer at the vegetation (‘tree’ SAOI). This can also be observed from the two transition graphs (Figures 2b and 2c). Looking at the color-coded circles, the SAOI with longer fixation duration is easily identified as the ‘tree’ SAOI for participant ‘po’. The focus of participant ‘zb’ on the car SAOIs is also depicted by the related transition lines in contrast for example to participant ‘kae’ that shows more diverse transitions between SAOIs related to cars, trees, and buildings. Comparing the two transition graphs, the SAOI radial transition graph in Figure 2b looks more compact but the clarity of the transition lines connecting adjacent radial sectors might be impaired. This issue is not present in the SAOI mirror radial transition graph in Figure 2c since all transition lines start from a sector on the left and reach one on the right.

The video example from the Eye Tracking Benchmark dataset is a 19 seconds video of a dialog scene between two people. We run the panoptic segmentation with a confidence threshold of 0.9. The algorithm extracted the correct SAOI classes with a few exceptions of some last frames for which it wrongly classified the wall as e.g. ‘sky’ or ‘snow’. This might be due to the simplicity of the scene and the lack of additional context. Moreover, the algorithm did not properly track the person on the right for the first two frames during the moment he enters the room, associating two different instance labels, ‘person2’ and ‘person3’, to the same person. The incorrect classes can be easily identified being at the bottom of the list which is ordered by percentage of the occupied area, from higher to lower percentage. Hence, SAOIs at the bottom of the list can be deselected and not taken into consideration for the analysis through the visu-

alizations. The scarf plot of six participants of the dialog video is shown in Figure 1a together with the interface of the tool. Due to page restrictions, Figures 1b-1f show the other available visualizations for only two participants, P2B and P14B, to assure a sufficient level of readability. Analyzing the SAOI radial and mirror radial transition graphs in Figures 1e and 1f, we can see different approaches between the two participants; while P2B has the focus evenly distributed between the two people of the video stimuli, P14B has more fixations on ‘person1’; moreover, P14B presents more transition lines. The same information could be gathered by analyzing the other visualizations one at a time, i.e. the fixation duration bar chart and the transition matrix, while the two radial transition graphs provide it in a single visualization. Moreover, the SAOI radial and mirror radial transition graphs also encode the size of the area of the SAOIs. This can be useful for the semantic analysis in case we want to compare the results between different SAOIs of the same class, e.g. ‘car4’ is much smaller than ‘car1’, hence we could consider normalizing the fixation data in this case (Holmqvist et al., 2015). Another case for which it could be useful to normalize the fixation data across SAOIs is when we compare different stimuli of the same scene, e.g. egocentric video from different participants using head-mounted devices.

## 5 DISCUSSION AND FUTURE WORK

The visual analytics made possible by this holistic semantic approach can be a useful and convenient method for an initial semantic analysis of the data when no other AOIs are defined yet. However, it is important to highlight that the methods relying on automatic semantic segmentation have to deal with possible errors in the classification. Even if the accuracy of the latest deep learning approaches is very high it needs to be considered when we build a visualization analysis upon these techniques. For image and short video stimuli, a visual check of the semantic segmentation can be enough; however, for longer video stimuli this needs to be handled in a different way, for example, by statistical filtering of the SAOIs outliers.

Panoptic segmentation provides the most comprehensive and distinctive type of segmentation and, at the same time, it is easy to convert its output in a more generic semantic segmentation by unifying all instances of the same class. This can be useful in long video stimuli or very complex image stimuli for which the differentiation of instances of a class might

result in the extraction of too many distinct SAOIs.

A limitation of the current implementation is that, in the case of video stimuli, we rely on the simple instance tracking available on Detectron2. Hence, in the case of dynamic SAOIs heavily overlapping with each other during the video we can lose track of the instances. Some recent works on video panoptic segmentation, e.g. (Kim et al., 2020) address this issue and we are planning to adopt similar solutions in our tool. At the present time, the use of deep learning approaches on a specific scenario requiring the training of the model might still be an obstacle due to the required large size of the training dataset. However, the advantages of this technique are numerous especially in the analysis of dynamic stimuli distinct among participants, such as head-mounted eye tracking data. In the case of natural stimuli covered by a large and reliable dataset such as MS COCO, the use of the pre-trained models available online allows a valid analysis if combined with the possibility to visually check and filter the segmentation output.

Regarding the proposed SAOI radial and mirror radial transition graphs, we plan to analyze their efficacy through a user study and to compare the two techniques. As future work, we also plan to explore the integration of more complex visualization techniques that handle for example hierarchical AOIs (Blascheck et al., 2016) by considering inter-class relationships between semantic classes. This would allow a multi-layer analysis of the SAOIs giving the possibility to the analyst to choose the granularity of the data. Another factor to consider is the scalability of the number of SAOIs processed by the tool. Since the SAOIs are color-coded, their total number needs to be limited to guarantee distinguishability between SAOIs both in the stimuli view and the visualization view.

## 6 CONCLUSIONS

We present an initial investigation on using panoptic segmentation for automatic extraction of semantic AOIs as a support for the analysis of eye tracking data through visualizations. Our visual analytics tool processes an image or a video dividing the entire stimulus on semantic AOIs and provides a set of AOI visualizations adapted to semantic AOIs. We propose a novel AOI visualization based on radial transition graphs. We show the capabilities of our tool by analyzing two application examples with data taken from online datasets. We plan to expand the analysis of our tool with further user evaluations and the implementation of other AOI-based visualization techniques.

## ACKNOWLEDGEMENTS

This work was supported in part by KK-stiftelsen Sweden, through the ViaTech Synergy Project (contract 20170056).

## REFERENCES

- Barz, M. and Sonntag, D. (2021). Automatic visual attention detection for mobile eye tracking using pre-trained computer vision models and human gaze. *Sensors*, 21(12).
- Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., and Ertl, T. (2017a). Visualization of eye tracking data: A taxonomy and survey. *Computer Graphics Forum*, 36(8):260–284.
- Blascheck, T., Kurzhals, K., Raschke, M., Strohmaier, S., Weiskopf, D., and Ertl, T. (2016). Aoi hierarchies for visual exploration of fixation sequences. ETRA '16, page 111–118, New York, NY, USA. Association for Computing Machinery.
- Blascheck, T., Raschke, M., and Ertl, T. (2013). Circular heat map transition diagram. ETSA '13, page 58–61, New York, NY, USA. Association for Computing Machinery.
- Blascheck, T., Schweizer, M., Beck, F., and Ertl, T. (2017b). Visual comparison of eye movement patterns. *Computer Graphics Forum*, 36(3):87–97.
- Borji, A. and Itti, L. (2013). State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):185–207.
- Burch, M., Kull, A., and Weiskopf, D. (2013). Aoi rivers for visualizing dynamic eye gaze frequencies. *Computer Graphics Forum*, 32(3pt3):281–290.
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2018a). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848.
- Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018b). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Fuhl, W., Kuebler, T., Brinkmann, H., Rosenberg, R., Rosenstiel, W., and Kasneci, E. (2018a). Region of interest generation algorithms for eye tracking data. ETVIS '18, New York, NY, USA. Association for Computing Machinery.
- Fuhl, W., Kuebler, T., Santini, T., and Kasneci, E. (2018b). Automatic Generation of Saliency-based Areas of Interest for the Visualization and Analysis of Eye-tracking Data. In Beck, F., Dachsbacher, C., and Sadlo, F., editors, *Vision, Modeling and Visualization*. The Eurographics Association.
- Goldberg, J. H. and Kotval, X. P. (1999). Computer interface evaluation using eye movements: methods and

- constructs. *International Journal of Industrial Ergonomics*, 24(6):631–645.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988.
- Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., and van de Weijer, J. (2015). *Eye tracking: a comprehensive guide to methods and measures*. Oxford University Press.
- Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95.
- Irimia, A., Chambers, M. C., Torgerson, C. M., and Van Horn, J. D. (2012). Circular representation of human cortical networks for subject and population-level connectomic visualization. *NeuroImage*, 60(2):1340–1351.
- Judd, T., Ehinger, K., Durand, F., and Torralba, A. (2009). Learning to predict where humans look. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2106–2113.
- Kim, D., Woo, S., Lee, J.-Y., and Kweon, I. S. (2020). Video panoptic segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9856–9865.
- Kirillov, A., Girshick, R., He, K., and Dollár, P. (2019a). Panoptic feature pyramid networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6392–6401.
- Kirillov, A., He, K., Girshick, R., Rother, C., and Dollár, P. (2019b). Panoptic segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9396–9405.
- Kurzhals, K., Bopp, C. F., Bässlér, J., Ebinger, F., and Weiskopf, D. (2014). Benchmark data for evaluating visualization and analysis techniques for eye tracking for video stimuli. In *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization*, BELIV '14, page 54–60, New York, NY, USA. Association for Computing Machinery.
- Kurzhals, K. and Weiskopf, D. (2013). Space-time visual analytics of eye-tracking data for dynamic stimuli. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2129–2138.
- Kurzhals, K. and Weiskopf, D. (2015). Aoi transition trees. In *Proceedings of the 41st Graphics Interface Conference*, GI '15, page 41–48, CAN. Canadian Information Processing Society.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T., editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.
- Panetta, K., Wan, Q., Rajeev, S., Kaszowska, A., Gardony, A. L., Naranjo, K., Taylor, H. A., and Agaian, S. (2020). Iseecolor: Method for advanced visual analytics of eye tracking data. *IEEE Access*, 8:52278–52287.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.
- Privitera, C. and Stark, L. (2000). Algorithms for defining visual regions-of-interest: comparison with eye fixations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):970–982.
- Raschke, M., Chen, X., and Ertl, T. (2012). Parallel scanpath visualization. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, page 165–168, New York, NY, USA. Association for Computing Machinery.
- Richardson, D. C. and Dale, R. (2005). Looking to understand: The coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. *Cognitive Science*, 29(6):1045–1060.
- Santella, A. and DeCarlo, D. (2004). Robust clustering of eye movement recordings for quantification of visual interest. In *Proceedings of the 2004 Symposium on Eye Tracking Research & Applications*, ETRA '04, page 27–34, New York, NY, USA. Association for Computing Machinery.
- Shelhamer, E., Long, J., and Darrell, T. (2017). Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651.
- Tighe, J., Niethammer, M., and Lazebnik, S. (2014). Scene parsing with object instances and occlusion ordering. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3748–3755.
- Tu, Z., Chen, X., Yuille, A. L., and Zhu, S.-C. (2005). Image parsing: Unifying segmentation, detection, and recognition. *Int. J. Comput. Vision*, 63(2):113–140.
- Wolf, J., Hess, S., Bachmann, D., Lohmeyer, Q., and Meboldt, M. (2018). Automating areas of interest analysis in mobile eye tracking experiments based on machine learning. *Journal of Eye Movement Research*, 11(6).
- Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019). Detectron2. <https://github.com/facebookresearch/detectron2>.
- Zanca, D., Serchi, V., Piu, P., Rosini, F., and Rufa, A. (2018). Fixatons: A collection of human fixations datasets and metrics for scanpath similarity.
- Zhang, H., Dana, K., Shi, J., Zhang, Z., Wang, X., Tyagi, A., and Agrawal, A. (2018). Context encoding for semantic segmentation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7151–7160.