sensors

MDPI

*Article*

# Automatic Visual Attention Detection for Mobile Eye Tracking Using Pre-Trained Computer Vision Models and Human Gaze

**Michael Barz [1,2,*] and Daniel Sonntag [1,2]**

[1] Interactive Machine Learning Department, German Research Center for Artificial Intelligence (DFKI), Stuhlsatzenhausweg 3, Saarland Informatics Campus D3_2, 66123 Saarbrücken, Germany; daniel.sonntag@dfki.de

[2] Applied Artificial Intelligence, Oldenburg University, Marie-Curie Str. 1, 26129 Oldenburg, Germany

* Correspondence: michael.barz@dfki.de

**Abstract:** Processing visual stimuli in a scene is essential for the human brain to make situation-aware decisions. These stimuli, which are prevalent subjects of diagnostic eye tracking studies, are commonly encoded as rectangular areas of interest (AOIs) per frame. Because it is a tedious manual annotation task, the automatic detection and annotation of visual attention to AOIs can accelerate and objectify eye tracking research, in particular for mobile eye tracking with egocentric video feeds. In this work, we implement two methods to automatically detect visual attention to AOIs using pre-trained deep learning models for image classification and object detection. Furthermore, we develop an evaluation framework based on the VISUS dataset and well-known performance metrics from the field of activity recognition. We systematically evaluate our methods within this framework, discuss potentials and limitations, and propose ways to improve the performance of future automatic visual attention detection methods.

**Keywords:** eye tracking; visual attention; eye tracking data analysis; area of interest; computer vision

check for updates

**Citation:** Barz, M.; Sonntag, D. Automatic Visual Attention Detection for Mobile Eye Tracking Using Pre-Trained Computer Vision Models and Human Gaze. *Sensors* **2021**, *21*, 4143. https://doi.org/10.3390/s21124143

Academic Editor: Jamie A. Ward

Received: 12 May 2021
Accepted: 12 June 2021
Published: 16 June 2021

## 1. Introduction

Eye tracking studies in many fields use Areas of Interest (AOIs) and visual attention to these AOIs as a common analytical helper tool. The resulting metrics are built to include events like AOI hits, dwells and transitions, which are based on raw gaze data or fixations with respect to a number of pre-defined AOIs. AOIs are tightly coupled to the hypotheses of a study because the corresponding metrics are used to argue for confirming or rejecting a hypothesis about visual stimuli. Hence, AOIs are very important, but incorrect placement of AOIs, and also inaccurate or imprecise mapping of gaze events to AOIs can heavily undermine the validity of a research study [1]. This adds the requirement of high robustness, accuracy, and precision for gaze estimation and gaze to AOI mapping methods. An AOI is usually defined as a spatial region with respect to the visual stimuli shown in a study, e.g., by defining a rectangular mask. For remote eye tracking with a static stimulus, it can be defined once and reused for every participant. The complexity increases if the stimulus is a video with dynamic AOIs, for example if they are linked to a dynamically moving object in the video. In this case, an AOI must be annotated for each video frame. This can be done frame-by-frame or, more efficiently, by defining bounding boxes for keyframes and interpolating intermediate frames [2]. Annotations can be reused if the video is the same for all participants: the participants' individual gaze or fixation points can be mapped to these AOI regions automatically. In mobile eye tracking studies, each recording comes with an individual video. Hence, AOI definitions using frame-wise and keyframe-based annotation approaches cannot be reused which makes them inefficient. An alternative is the fixation-wise annotation: per fixation, an annotator has to decide whether an AOI is hit or not based on the visual stimulus around the fixation point [1]. A typical fixation lasts around 200–400 ms, which reduces the annotation effort compared to a frame-wise annotation. However, fixation-wise annotation does not remedy the need to annotate AOIs in every recording of every participant. A solution can be found in attaching fiducial markers to, e.g., a target stimulus in 2D [3] and 3D [4], an interactive area [5], or tangible objects [6]. In this research, we aim at circumventing the requirement to instrument the environment with obtrusive markers.

Previous methods tackled the automatic analysis of head-mounted eye tracking data in uninstrumented environments [7–16]. Drawbacks of these methods include, e.g., a missing support for real-time applications, and the restriction to a limited number of classes (≤12). Further, not all papers report quantitative evaluation results [7,8,14] or do not properly describe their evaluation metrics [9,10], or use inadequate metrics that ignore temporal aspects [12]. Some commercial tools offer automatic mapping of the gaze signal in world video coordinates to a reference frame that defines AOIs (For example, see the assisted mapping function from Tobii Pro, accessed on 15 June 2021). However, this is only possible for a limited number of reference frames.

In this work, we implement two methods for automating the detection of attention to task-related objects or AOIs in real-time. This can help in analyzing complex interaction behavior of humans: it bears the potential to facilitate novel real-time adaptive human-computer interaction [7,17], and to boost the efficiency in research based on eye tracking by automating the time-consuming and expensive data annotation process [12]. We contribute by (i) implementing two methods for detecting visual attention using eye tracking data and pre-trained deep learning models for image classification and object detection; and (ii) evaluating the performance of our methods using the VISUS dataset [2] and fine-grained activity recognition metrics in a systematic way [18]. The proposed evaluation framework and our results should serve as a reference for upcoming methods in automatic gaze to AOI mapping. Further, we discuss the performance of our methods and how interactive transfer learning can be used to break the limitations of pre-trained models.

## 2. Related Work

Our work aims at accelerating and objectifying research on visual attention with mobile eye tracking using technologies from the field of computer vision. In human perception, "selective visual attention is the allocation of limited attentional resources to certain information in the visual field, while ignoring other information" [1] (p. 26). It can be guided by salient bottom-up factors and task-related top-down factors in a scene [19]. When humans perform a task, the number of fixations to irrelevant but salient objects drop, while the fixations to task-relevant objects, i.e., top-down factors, increase [1,20–23]. In the following, we summarize related work that used human gaze for intelligent human-computer interaction, and we describe related approaches that addressed the problem of automatic or semi-automatic gaze-to-AOI mapping in non-instrumented environments. Further, we provide a brief overview on the state-of-the-art in computer vision in this regard.

### *2.1. Eye Gaze in Human-Computer Interaction*

Human gaze, which can be seen as a proxy for human visual attention, can be beneficial when applied in intelligent human-machine interaction [24–26]. It can be used as an active or passive input modality [27]. For instance, a user can influence a system via explicit eye movements (active) and a system can implicitly derive information about the user, its state, and intentions by observing the eye movement behavior (passive). In this paper, we focus on eye gaze as an implicit source of context information. Related works in this field investigated and applied eye gaze in the context of conversational interfaces, information retrieval systems, and situation-aware human-machine interaction, in general.

Ishii et al. [28] proposed a system for estimating the user's conversational engagement using eye tracking data from a Wizard-of-Oz study. In a subsequent work, they modelled turn-taking behavior in human-human dialogues based on eye gaze features [29]. A similar approach was presented by Jokinen et al. [30]. Prasov and Chai [31] developed a system that combines speech and eye gaze to enhance reference resolution in conversational interfaces.

Xu et al. [32] investigated the role of mutual gaze in a human-robot collaboration setting and found that maintaining eye contact leads to improved multimodal interaction behavior of users, i.e., more synchronized and coordinated. Baur et al. [33] implemented NovA, a system for analyzing and interpreting social signals in multi-modal interactions with a conversational agent, which integrates eye tracking technology. Thomason et al. [34] developed a gaze-based dialog system that enables grounding of word meanings in multimodal robot perception.

In the domain of information retrieval, Buscher et al. [35] investigated the relation between reading behavior and document relevance. The authors introduced the concept of attentive documents that keep track of the perceived relevance based on eye movements. Other works investigated the utility of eye tracking in multimedia retrieval settings. Several algorithms were proposed for estimating the search target of an ongoing visual search on a screen [36–39]. Barz et al. [40] introduced an algorithm for estimating the target segment of a visual search in more natural settings.

Eye tracking was also used to facilitate situation-aware human-machine interaction in general. Bulling et al. [41] presented an approach for inferring high level contextual cues from eye movements to facilitate behavioral monitoring and life-logging. Similarly, Steil and Bulling [42] used topic modeling to detect everyday activities from eye movements in an unsupervised fashion. In a later work, the authors presented an approach for visual attention forecasting in mobile interaction settings which takes the visual scene and device usage data as additional inputs [43]. Also, other works combine visual features of a scene with gaze information for recognizing recent actions [44–47]. In the context of human-robot interaction, Ramirez-Amaro et al. [48] showed that human behavior inference benefits from incorporating mobile eye tracking data with third person videos. Recently, Kurzhals et al. [49] described an interactive approach for annotating and interpreting egocentric eye tracking data for activity and behavior analysis. They implement an iterative time sequence search based on eye movements and visual features. Steichen et al. [50] investigates the effectiveness of eye tracking for predicting user characteristics like cognitive abilities and the utility of such a model in adaptive information visualization. A comparison of uni-modal and multi-modal methods for user modeling in the context of real-time adaptive data visualization can be found in [51]. These works aim at segmenting eye tracking recordings into phases of different activities. Our goal is to identify phases of attention to objects in a scene that can serve as AOI.

### 2.2. Gaze-to-AOI Mapping

A few works address the problem of mapping human gaze to objects or areas of interest in non-instrumented environments. Pontillo et al. [10] presented SemantiCode, an interactive tool for post-hoc fixation-based annotation of egocentric eye tracking videos. It supports semi-automatic labelling using a distance function over color histograms of manually annotated fixations. Brône et al. [14] proposed to use object recognition with mobile eye tracking to enhance the analysis of customer journeys. In follow-up works, they compared different feature extraction methods [52] and evaluated their approach in a museum setting [15]. Evans et al. [53] reviewed methods for mobile eye tracking in outdoor scenes ranging from pupil detection and calibration to data analysis. They presented an early overview of methods for automating the process of analyzing mobile eye tracking data. Fong et al. [54] presented a semi-automatic data annotation approach using on the human-in-the-loop principle. The annotator can label individual frames based on the visual appearance and the gaze position. As the annotation process advances, the system learns the appearance of AOIs based on examples and to automatically classify clearly similar cases. Panetta et al. [12] presented an annotation method based on bag-of-visual-words as features and a support vector classification model (SVC) that is trained before the analysis takes place. In a follow-up work, the authors present a system that automatically segments objects of interest using two state-of-the-art neural segmentation models [55]. They use pre-trained models to showcase and evaluate new data visualization methods, but they did not assess the performance of their automatic annotation approach. Kurzhals et al. [8] used bag-of-SIFT features and color histograms with unsupervised clustering to sort fixation-based image patches by their appearance. They offer an interactive visualization for manual corrections. Venuprasad et al. [13] use clustering with gaze and object locations from an object detection model to detect visual attention to an object or a face. Sümer et al. [56] investigated the problem of automatic attention detection in a teaching scenario. They extract image patches for all student faces in the egocentric video feed and cluster them using activations from a ResNet-50 [57] model trained on VGGFace2 data [58]. They assign student IDs to each cluster which allows them to map the teacher's gaze to individual students. Callemein et al. [59] presented a system for detecting when the participant's gaze focuses the head or hands of another person without the possibility to differentiate between interlocutors. Other works also focused on real-time applications. For example, Toyama et al. [11] implemented the Museum Guide that uses SIFT (scale-invariant feature transform) features [60] with the nearest neighbor algorithm and a threshold-based event detection to recognize user attention to one of 12 exhibits. They extended their approach to detect read texts and fixated faces with the goal to build artificial episodic memories to support dementia patients [61]. Barz and Sonntag [7] presented a similar approach using a GoogLeNet model [62] pre-trained with ImageNet [63] data. Wolf et al. [16] implemented the computational Gaze-Object Mapping algorithm that maps fixations to object-based AOIs using the Mask R-CNN object detection model [64]. They conducted a controlled lab study to record data in a healthcare setting with two AOIs, a *bottle* and five *syringes*. An evaluation has shown that, using 72 training images with 264 annotated object masks, their system can closely approximate the AOI-based metrics in comparison to manual fixation-wise annotations as a baseline. Batliner et al. [65] presented a similar system for simplifying usability research with mobile eye trackers for medical screen-based devices. Machado et al. [9] matched fixations with bounding boxes of another object detection algorithm to detect user attention. They used a sliding-window approach with a MobileNet model [66], pre-trained on ImageNet data, to detect objects in an image.

### *2.3. Computer Vision*

Computer vision is the algorithmic equivalent to human visual perception and subsumes image classification and object detection methods. Image classification refers to the assignment of a single label to an image, object detection refers to the localization and classification of multiple objects in a single image [63]. Recent methods experienced a performance boost with the advance of deep learning technology and the availability of large datasets for model training. Popular examples are the ImageNet dataset for image classification [63] and the MS COCO dataset for object detection [67]. A recent overview of object detection with deep learning can be found in [68]. We apply residual network models introduced in [69], pre-trained on ImageNet, and the Mask R-CNN model for object detection [64], pre-trained on MS COCO. Related works also include methods for egocentric activity recognition without gaze data. For example, Ma et al. [70] use hand segmentations, object localizations and the optical flow from first-person videos to infer ongoing activities. Another example is EgoNet by Bertasius et al. [71] which determines the action-object in egocentric videos.

## 3. Method

We implement two methods for an automatic detection of visual attention to a visual stimulus in a scene. Both take the video feed and the corresponding gaze or fixation signal as input and predict, if the participant paid attention to an AOI for each frame (see Figure 1). The first method, *IC*, aggregates classifications of image patches, cropped around the gaze signal, using a pre-trained image classification model similar to the gaze-guided object classification system by [7]. The second method, *OD*, matches fixation events with the result of a pre-trained object detection model similar to Wolf et al. [16] and Machado et al. [9]. In this work, we concentrate on pre-trained computer vision models, similar to Barz and Sonntag [7] and Machado et al. [9], to explore when models without a training overhead can be applied effectively and when they reach their limits. We leave fine-tuning of the models as a task for future-work, because it is outside the scope of this paper. Both methods are implemented in Python using the *multisensor-lpipeline* (https://github.com/DFKI-Interactive-Machine-Learning/multisensor-pipeline, accessed on 15 June 2021) package for flexible streaming and processing of signals from one or multiple sources. It allows to easily set up real-time applications using source modules for connecting sensor input, processor modules for manipulating or aggregating incoming data streams and events, and sink modules for, e.g., storing and visualizing the output. In the following, we describe the implementation of both methods and their adjustable parameters.

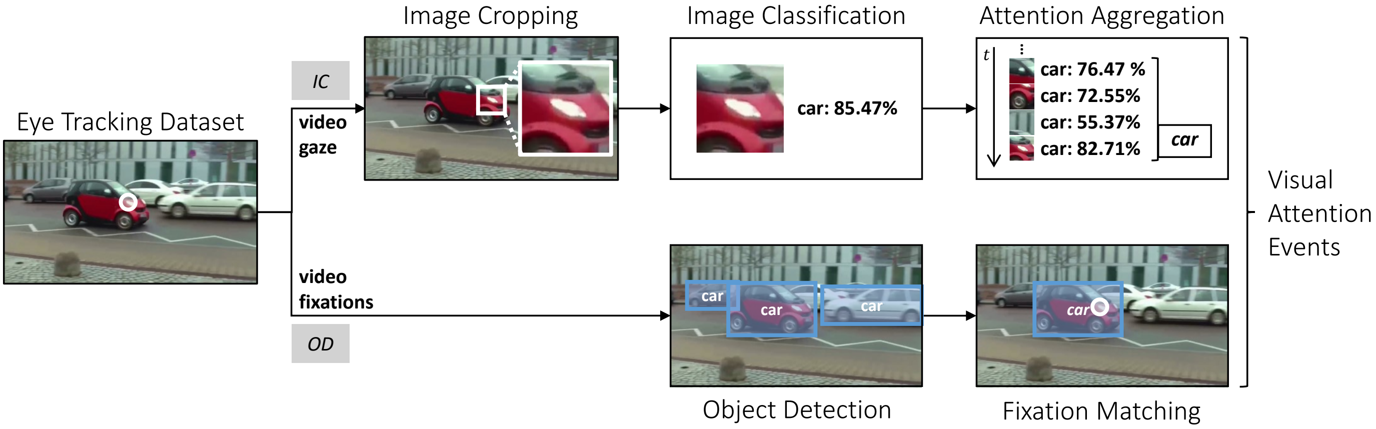


**Figure 1.** Processing workflow of the two proposed methods for automatic attention detection: *IC* is based of image classification and gaze samples, *OD* uses object detection and fixation events. Both methods support visual attention detection in real-time.

### 3.1. Detect Attention Using Gaze-Guided Image Classification (IC)

Our method based on image classification includes four subsequent steps. First we re-sample the gaze signal to 5 Hz and crop an image patch of 200 × 200 pixels from the egocentric video feed (1920 × 1080) per remaining sample. We use this crop size, because it turned out to perform well in real-time applications (see [7,72]) and the size fits well to the AOIs in the VISUS dataset (manual inspection). Second, each patch is classified using a pre-trained version of the ResNet image classification model [69] which is trained on the ImageNet dataset with 1001 object classes [63]. The prediction result includes the top-5 class candidates and their probability. In a third step, we aggregate same or similar class labels by accumulating their probabilities. We merge similar object classes based on a manually defined lookup table. For example, if the top-5 output includes the ImageNet classes `passenger car`, `streetcar` and `limousine`, we replace the probability of `passenger car` by the sum of all three probabilities and remove the remaining class labels from the output. In the last step, we implement a working memory- and threshold-based attention detection algorithm similar to Barz and Sonntag [7] and Toyama et al. [11] using the top-1 predictions of the previous step as continuous input:

An update routine is called for each incoming prediction, i.e., a tuple including a unique class label and the corresponding probability output of the model: $(c, p(c))$. If $p(c)$ exceeds the minimum probability $T_p$, we increase the duration counter $C_{dur}$ at the index $c$ by the amount of milliseconds that passed since the last run of the update loop (circa 200 ms). For all other classes with a non-zero duration count, we increase the noise counter $C_{noise}$ by the same amount of time. If the aggregated duration $C_{dur}[c]$ exceeds the duration threshold $T_{dur}$, we send an *attention started event* including $c$, $p(c)$ and the timestamp of the latest prediction. In addition, we store $c$ as the currently attended class $c_{active}$ and reset both counters for it: we set $C_{dur}[c]$ and $C_{noise}[c]$ to zero. If

$c_{active}$ is not empty and not equal to $c$, we consider the prior attention event to be over and send an *attention ended event*. We send an *attention confirmed event* is $c$ is equal to $c_{active}$. Finally, we check for all remaining classes whether the aggregated noise duration in $C_{noise}$ exceeds the noise threshold $T_{noise}$. In this case, we reset both counters for this class and, if this class is equal to $c_{active}$, we send an *attention ended event*. We subtract $T_{dur}$ from the event timestamp to better match the actual start and end times of the attention events. We offer a parameter for setting the image classification *model*: we include the ResNet-50 and ResNet-152 models via Tensorflow Hub. The pre-trained models can be found at https://tfhub.dev/google/imagenet/resnet_v2_50/classification/4 and https://tfhub.dev/google/imagenet/resnet_v2_152/classification/4, respectively (each accessed on 2 May 2021). Any other model from this platform, that was trained using ImageNet, can be used as well by providing a corresponding link. The default setting is $T_{dur} = T_{noise} = 300$ ms, $T_p = 40\%$, and the *model* is set to ResNet-152. We refer to this setting as IC-152-300-40 (in general: IC-*model*-$T_{dur/noise}$-$T_p$).

### 3.2. Detect Attention Using Object Detection (OD)

Our second method is based on an object detection model which can detect multiple object instances in an image from a set of candidate classes. To detect visual attention, we match the position of fixation events from the eye tracker with detected object regions. For each fixation, we extract an image frame from the video feed that is closest to the start of the fixation event. The object detection takes longer per image than the image classification algorithm. However, this method can still be applied in real-time, because it is applied once per fixation. During a fixation the eye is relatively still and, hence, should point to the same location in the world space. But, fixation detection is not perfect, e.g., in presence of smooth pursuit movements, which makes this method dependent on the quality of the applied fixation detection algorithm. Next, we detect all object instances in the current image frame: we use a Mask R-CNN model [57] that is pre-trained on the MS COCO dataset [67] with the Detectron2 framework [73]. The model weights can be downloaded from https://dl.fbaipublicfiles.com/detectron2/COCO-InstanceSegmentation/mask_rcnn_R_101_FPN_3x/138205316/model_final_a3ec72.pkl (accessed on 2 May 2021). For each instance, it provides a class label with a probability value, as well as a rectangular bounding box and a pixel-wise segmentation mask depicting the object area. Finally, we check whether the fixation position lies within the object area, either using the bounding boxes (*bbox*), similar to Machado et al. [9], or the more fine-grained segmentation masks (*mask*), similar to Wolf et al. [16], as reference. This can be configured via the *object mask* parameter that defaults to *bbox*. If a hit is detected, we send an *attention started event* using the start time of the fixation and an *attention ended event* using its end time. If two object areas are hit, we choose the one with higher probability. We refer to the two possible settings as OD-bbox (default) and OD-mask.

## 4. Evaluation

We evaluate the performance of the two methods described above in terms of their ability to detect time intervals in which a participant fixates a certain AOI. Our evaluation procedure utilizes the VISUS dataset [2] including eye tracking data from 25 participants for 11 scenarios, and manual AOI annotations which we use for ground truth extraction. To measure the performance, we use a set of frame- and event-based metrics by Ward et al. [18] from the field of activity recognition which allow a more fine-grained analysis. We report the metrics per scenario and for each of the 34 AOIs to identify effective applications and limitations.

### 4.1. Dataset

We use the VISUS dataset for our evaluation [2] which can be downloaded from https://www.visus.uni-stuttgart.de/publikationen/benchmark-eyetracking (accessed on 12 April 2021). It contains eye tracking data from 25 participants for 11 video stimuli, totaling to 275 sessions. The gaze data was recorded using a Tobii T60 XL remote eye tracker at 60 Hz. The authors did not report the spatial accuracy and precision as measured during their recordings. The video stimuli have a resolution of 1920 × 1080 pixels at 25 frames per second and have an average length of 75.55 s ($SD = 59$). Each video is manually annotated with axis aligned rectangular bounding boxes from two annotators for 1 to 6 AOIs per video (see Table 1). Bounding boxes were set at key frames and interpolated for intermediate frames. The main purpose of the dataset is to serve as a benchmark for visualization and analysis techniques in the field of eye tracking. We use the dataset as a benchmark dataset for automatic detection of visual attention to dynamic AOIs. We treat the fixation events reported in the dataset that hit the manually defined bounding boxes as ground truth attention events to the respective AOIs. If two AOIs in a single frame are hit, we select the AOI that yields the longer event. While the VISUS dataset is acquired with a remote tracking device, we use it to approximate mobile eye tracking recordings: we do not leverage that the videos are the same for each participant. In the following, we describe the ground truth extraction, the scenarios (video stimuli) and AOIs, and we describe the related challenges for gaze to AOI mapping.

**Table 1.** Overview of scenarios and AOIs in the VISUS dataset and the corresponding mappings of class labels to AOIs. Class labels originate from ImageNet in case of *IC* methods and from MS COCO in case of *OD* methods.

| Scenario | AOI | ImageNet Labels | MS COCO Labels |
|---|---|---|---|
| `01-car pursuit` (25 s) | red car | streetcar, sports car, minivan, cab, minibus, limousine, car mirror, racer, passenger car | car |
| | white car | – | – |
| `02-turning car` (28 s) | red car | streetcar, sports car, minivan, cab, minibus, limousine, car mirror, racer, passenger car | car |
| `03-dialog` (19 s) | left face | ear | person |
| | right face | – | – |
| | shirt | sweatshirt | – |
| `04-thimblerig` (30 s) | cup1 | cocktail shaker, coffee mug, cup | cup |
| | cup2 | – | bowl |
| | cup3 | – | – |
| `05-memory` (148 s) | cards | desk | dining table |
| `06-UNO` (121 s) | left hand | – | person |
| | right hand | – | – |
| | stack covered | desk | dining table |
| | stack uncovered | – | – |
| `07-kite` (97 s) | person | lab coat, poncho, cardigan, cloak, sweatshirt, trench coat | person |
| | kite | balloon, kite, parachute | kite |
| `08-case exchange` (27 s) | persons | sombrero, cowboy hat | person |
| | textbox | – | – |
| | case | mailbag, packet, plastic bag, shopping basket, backpack, bucket, crate | handbag, suitcase |
| | suspects | lab coat, poncho, cardigan, cloak, sweatshirt, trench coat | – |
| `09-ball game` (31 s) | ball | baseball, basketball, rugby ball, tennis ball, volleyball, soccer ball | sports ball |
| | player white | ballplayer | person |
| | player red1 | – | – |
| | player red2 | – | – |
| | player red3 | – | – |
| `10-bag search` (133 s) | red bag | plastic bag | handbag |
| | yellow bag | – | – |
| | blue bag | – | – |
| | red-white bag | – | – |
| | brown bag | mailbag | – |
| | persons | lab coat, poncho, cardigan, cloak, sweatshirt, trench coat | person |
| `11-person search` (172 s) | hooded | lab coat, poncho, cardigan, cloak, sweatshirt, trench coat | person |
| | red shirt and hat | sombrero, cowboy hat | – |
| | persons | – | – |

#### 4.1.1. Scenarios & Challenges

The dataset includes 11 scenarios each with a different kind and number of AOIs. They pose multiple challenges to attention detection methods. In the simplest case, a method has to map gaze to AOIs that represent distinct concepts (challenge I). This applies to, e.g., `01-turning car` in which a single AOI, a "red car", is shown, and to `07-kite` with two distinct AOIs: a "person" flies a "kite". The difficulty increases, if two AOIs in a scenario refer to the same concept (challenge II). For instance, the scenario `01-car pursuit` shows a "red car" driving through a turning area, with a "white car" on the opposing lane and multiple parking cars in the background. The challenge is not only to detect that a car is fixated, but to differentiate between the two prominent cars (AOIs) and the background cars which are multiple instances of the same concept. Similarly, the scenarios 03, 08, 09, and 11 require the ability to differentiate between multiple instances of the concept person, for instance, a "hooded" person, a person wearing a "red shirt and hat", and several distractor "persons" in scenario `11-person search`. The problem complicates, if two AOIs not only share a concept, but also their appearance (challenge III). An example can be found in scenario `04-thimblerig` which includes three cups with identical appearance. Distinguishing them requires object tracking for multiple instances and, hence, an initial assignment of each instance to an AOI by hand. The scenario `05-memory` is not covered by the aforementioned cases. It shows a memory game: in the beginning, all 16 "cards" look the same, while, until the end of the game, we see 8 pairs of cards with different visual appearance per pair. Yet, all "cards" count toward the same AOI. The challenge is, if the appearance of an AOI changes over time (challenge IV).

#### 4.1.2. Mapping Class Labels to AOIs

Our methods aim at solving the aforementioned challenges using pre-trained computer vision models. For this, AOIs need to be mapped to class labels of ImageNet for the *IC* method and of MS COCO for the *OD* method. We assume that the performance per scenario depends on the type of AOIs and whether they are represented in the training data of the model. If there is no matching class label for an AOI, none of the methods can detect respective attention events. If a class label matches multiple AOIs of a scenario, i.e., if they share a concept, we can only assign the label to one of them. This probably leads to an increase in false positives. The performance might also suffer from inadequate matches. For this experiment, we use a separate mapping from class labels to AOIs for each method and scenario, as shown in Table 1. For *IC* methods, we identified ImageNet labels for 19 AOIs including adequate matches like *passenger car* for the AOI "red car", but also weak matches like *sweatshirt* as a proxy for the AOI "person". Similarly, we found MS COCO labels for 17 AOIs for the *OD* methods. For instance, *car* is an adequate match for the AOI "red car", while *dining table* is a weak match for the "stack covered" in `06-UNO` (the stack is located on a table).

### *4.2. Metrics*

To quantify the performance of our methods, we need evaluation metrics that depict how well our detected attention events match the ground truth events. We reviewed the metrics proposed in closely related works, but none of them was fully satisfactory: Panetta et al. [12] compared their system to manual ground truth annotations by calculating the distance between two histograms that aggregate the duration of fixations from predicted or ground truth AOI regions, respectively. However, their metric does not punish if detected AOI fixations are shifted in time or if they occur in the wrong order, which puts the validity of their metric into question. For instance, the histogram would be equal, if the predicted events were reported reversely. Machado et al. [9] reported accuracy and precision, but it is unclear whether they compute the metrics frame-wise or event-based. Toyama et al. [11] reported event-based precision and recall for each method: precision reflects how many of the detected attention events were classified correctly, recall indicates the proportion of detected attention events to all attention events. Similarly, De Beugher et al. [15] reported precision and recall, but at the frame-level. Wolf et al. [16] and Batliner et al. [65] reported the recall (true positive rate) and the specificity (true negative rate) at the frame-level, including one frame per fixation for the analysis. The specificity reflects the ratio of frames that are correctly classified as not showing human attention to an AOI (negative class) in relation to all frames with a negative class label. Sümer et al. [56] compared the absolute number of predictions for each class, i.e., four individual students, to the ground truth count. In addition, they use a confusion matrix to show the performance of their face recognition system that is used to assign fixations to students' faces. Callemein et al. [59] used measures for inter-rater agreement like Cohen's $\kappa$ to show the performance of their gaze-to-face and gaze-to-hand mapping. Venuprasad et al. [13] reported precision, recall, and accuracy for frames, and event metrics based on detection events: first looks, extra looks (i.e., revisits), false positive and false negative events are counted. Other works reported qualitative results only or did not evaluate their method. In this work, we report fine-grained frame- and event metrics per AOI from the field of activity recognition [18]. They were shown to be effective for evaluating event detection methods in the field of mobile eye tracking [18,74]. The metrics are based on a segmentation of the ground truth and prediction signal at the frame level per AOI (see Figure 2). A segment ends, if the ground truth or the prediction changes, i.e., both signals are constant within a segment. Each segment can now be rated as one of true positive, true negative, false positive, or false negative. The event and frame metrics are derived from these segments. Prior to feature computation, we remove events with a duration smaller than the frame time and merge adjacent events.

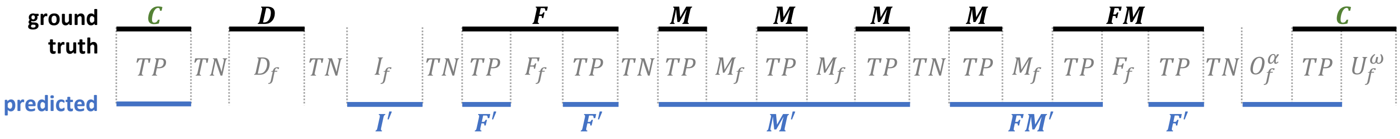


**Figure 2.** Example of segmented ground truth events and predicted events with annotations for event error and frame error classes. The vertical bars depict the segment boundaries. The frame error classes are given per segment.

### 4.2.1. Event Metrics

Ward et al. [18] define a set of error classes for events which are meant to characterize the performance of a single-class event detection method. For multi-class problems, each class is handled separately. Error classes include the insertion ($I'$) and deletion ($D$) error which are commonly used in event detection. An insertion error depicts that a detected event is not present in the ground truth (false positive), and a deletion error indicates a failure in detecting a ground truth event (false negative). Additional error classes include fragmentation and merge errors: a ground truth event is fragmented ($F$), if multiple fragmenting events ($F'$) are detected in the output. Similarly, multiple ground truth events of the same class can be merged ($M$) by a single merging event ($M'$) in the output. Both errors can appear together, e.g., if a ground truth event is fragmented by three event detections of which the third is merging an additional ground truth event. In this case, the first ground truth event is marked as fragmented and merged ($FM$), and the third event detection is marked as fragmenting and merging ($FM'$). The apostrophe indicates whether an error class is assigned to a ground truth event or a predicted event in the output. If none of the error classes can be assigned, a detected event is counted as correct (C), i.e., as a true positive. According to Ward et al. [18], we visualize the metrics by means of an event analysis diagram (EAD). It shows the number and ratio of error classes in relation to the number of reference events, i.e, to the number of ground truth events $|E| = D + F + FM + M + C$, the number of predicted events (or returns) $|R| = M' + FM' + F' + I' + C$, or both in case of correct predictions $C$. Also, we can compute event-based precision and recall as a ratio between $|R|$ or $|E|$ and the error class counts. We compute a conservative precision as $Pr = \frac{C}{|R|}$ and recall as $Re = \frac{C}{|E|}$. Counting $F, FM, M$ and $M', FM', F'$ as correct, similar to Toyama et al. [11], we calculate a more progressive precision as $Pr^* = \frac{|R|-I'}{|R|}$ and recall as $Re^* = \frac{|E|-D}{|E|}$.

### 4.2.2. Frame Metrics

For extracting the frame metrics, Ward et al. [18] project error classes to frames per segment. Similar to event-based error classes, a frame can be rated as insertion ($I_f$), deletion ($D_f$), merge ($M_f$), or fragmentation ($F_f$). Merge errors are assigned to false positive frames from merging events and fragmentation errors are assigned to false negative frames between fragmenting events. Further, if a neighboring segment is classified as true positive, frames of a false positive segment are marked as overfill ($O_f$) and frames of a false negative segment are marked as underfill ($U_f$). In other words, an overfill occurs, if a detected event starts early or ends late, and an underfill occurs, if a detected event starts late or ends early. A superscript indicates whether an underfill or overfill occurs at the start ($\alpha$) or end ($\omega$) of an event. Frames of true positive ($TP$) and true negative ($TN$) segments are classified likewise. Ward et al. [18] define the frame metrics as ratios of the error class counts and the total positive frames $P$ or negative frames $N$ in the ground truth, with $P = D_f + F_f + U_f^\alpha + U_f^\omega + TP$ and $N = I_f + M_f + O_f^\alpha + O_f^\omega + TN$. The resulting ratios (lowercase equivalents to error classes) can be used to express the false positive rate as $fpr = ir + mr + o^\alpha + o^\omega$, and one minus the true positive rate as $(1 - tpr) = dr + fr + u^\alpha + u^\omega$. We use a set of two stacked bar charts to visualize the frame metrics (compared to pie charts in [18]).

### 4.3. Experiment Conditions & Procedure

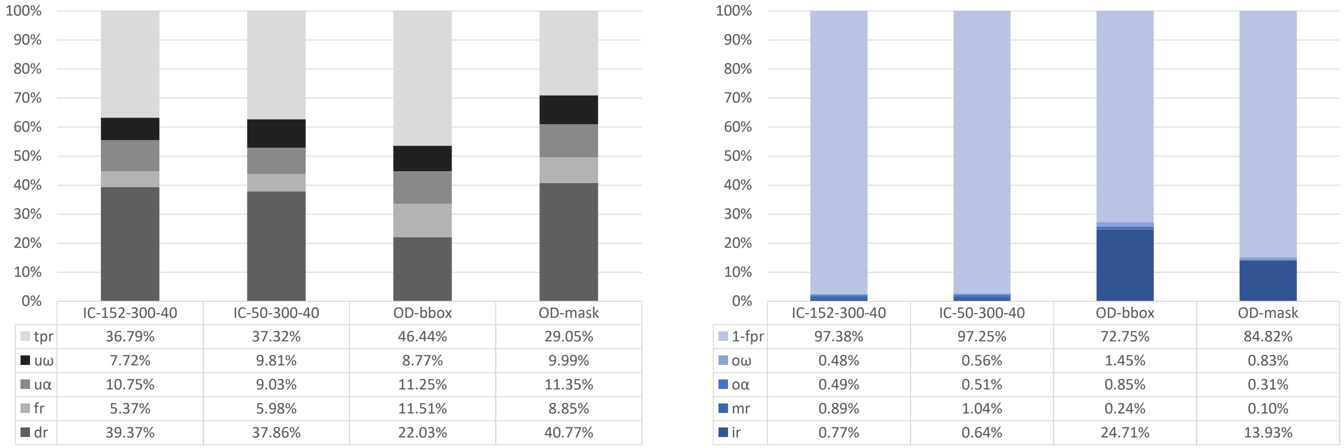
We compare two methods for visual attention detection: *IC* based on gaze-guided image classification and a threshold-based event detection, and *OD* based on object detection and fixation mapping. We generate predictions for the VISUS dataset using each method and analyze their results. We start with default parameters to identify AOIs which are not supported. We define cases with a recall of zero to be failing: this corresponds to a deletion rate of $dr = 100\%$ (frame metrics), or if all ground truth events are marked as deletions $D$. By design, we expect AOIs without a matching class label to fail (see dashes in Table 1). For the remaining AOIs, we investigate the impact of different methods and parameters on the performance metrics. We compare two *IC* methods using the classification *models*, ResNet-50 and ResNet-152, and two *OD* methods using the *object mask* options bbox and mask. The other parameters are set to their defaults, which results in the following set of parameterized methods: IC-152-300-40, IC-50-300-40, OD-bbox, and OD-mask. For *IC*, we additionally test different values for $T_{dur}$, $T_{noise}$, and $T_p$ using the ResNet-152 *model*, with $T_{dur} = T_{noise} \in \{100, 300, 500, 700\}$ ms and $T_p \in \{20\%, 40\%, 60\%\}$. Changing these parameters might have an effect on the performance of the *IC* method. Per method, we compute the frame and event metrics for each AOI and per participant. We sum the metrics over participants, if we report the performance per AOI, and over participants and AOIs, if we report the overall performance of a method. Summing the metrics corresponds to concatenating the recordings of all participants per AOI, because the metrics are based on absolute counts. Ratios are computed afterwards using the number of positive and negative ground truth frames or events which we add up as well.

### 4.4. Results

Using default parameters, we observe a recall of zero for all AOIs without a matching class label for a method, but also for other AOIs: for the *IC* method, this includes "left face", "cup1", "cards", "stack covered", "case", "player white", "red bag", "brown bag", and "hooded". For the *OD* method, this includes "cup1", "cup2", "cards", "stack covered", "case", "red bag", and "persons" (for `10-bag search` only). We count one additional AOI for *OD* ("ball") and three AOIs for *IC* ("suspects", "ball", and "persons" in `10-bag search`) as failing, because they yield a recall close to zero ($dr \geq 90\%$). The remaining six AOIs for *IC* and nine AOIs for *OD* are analyzed in detail (AOIs are listed in Section 4.4.2). The AOIs for *IC* include $|E|$ = 2438 ground truth events that correspond to $P$ = 71,911 positive frames and $N$ = 111,467 negative frames. The AOIs for *OD* include $|E|$ = 4328 events with $P$ = 154,783 positive and $N$ = 270,750 negative frames.

### 4.4.1. Overall Performance

We compare the metrics of two *IC* and two *OD* methods that vary in terms of the *model* or *object mask* setting (see Section 4.3). The frame metrics for remaining AOIs are summarized in Figure 3. It shows the ratios of false negative errors with respect to $P$ in Figure 3a, and of false positive errors with respect to $N$ in Figure 3b. Concerning the false negative errors, deletion is the most prominent class across all methods: they account for 22.03% (OD-bbox) and 40.77% (OD-mask) of the errors for *OD*, and $dr$ is 39.37% for ResNet-152 and 37.86% for ResNet-50 for the *IC* methods. On average, the $tpr$ does not differ between *OD* (37.75%) and *IC* (37.05%). However, OD-bbox yields the best $tpr$ with 46.44%, which is 9.39% better than the average of both *IC* methods and 17.39% better than OD-mask. The remaining error classes account for 30.86% for *OD* and 24.33% for *IC*: on average, *IC* faces 6.53% less false negatives through fragmenting events and underfills than *OD*. Concerning the false positive errors (Figure 3b), insertions are most prevalent for *OD* with $ir = 24.71\%$ for OD-bbox and $ir = 13.93\%$ for OD-mask. For *IC*, we observe less insertions, averaging to 0.71%. Errors from merging event detections and overfills account for 1.98% (*IC*) and 1.89% (*OD*). Hence, the $fpr$ adds up to 2.69% for *IC* and to 21.21% for *OD*, which means that the *OD* methods cause 18.53% more false positive errors at the frame level, on average.



| | IC-152-300-40 | IC-50-300-40 | OD-bbox | OD-mask |
|---|---|---|---|---|
| ■ tpr | 36.79% | 37.32% | 46.44% | 29.05% |
| ■ uω | 7.72% | 9.81% | 8.77% | 9.99% |
| ■ uα | 10.75% | 9.03% | 11.25% | 11.35% |
| ■ fr | 5.37% | 5.98% | 11.51% | 8.85% |
| ■ dr | 39.37% | 37.86% | 22.03% | 40.77% |

(**a**) false negative errors

| | IC-152-300-40 | IC-50-300-40 | OD-bbox | OD-mask |
|---|---|---|---|---|
| ■ 1-fpr | 97.38% | 97.25% | 72.75% | 84.82% |
| ■ oω | 0.48% | 0.56% | 1.45% | 0.83% |
| ■ oα | 0.49% | 0.51% | 0.85% | 0.31% |
| ■ mr | 0.89% | 1.04% | 0.24% | 0.10% |
| ■ ir | 0.77% | 0.64% | 24.71% | 13.93% |

(**b**) false positive errors

**Figure 3.** Frame metrics with respect to positive (**a**) and negative (**b**) ground truth frames across all AOIs.

Further, we report event metrics which are normalized by the number of ground truth events $|E|$ or the number of retrieved events $|R|$ (see Figure 4). Both *IC* methods show a similar distribution of error classes. For IC-152-300-40, we observe a high fraction of deletions, $\frac{D}{|E|} = 66.41\%$, and a low fraction of insertions, $\frac{I'}{|R|} = 5.99\%$, which is consistent to frame metrics. 371 predictions are correct which corresponds to $Re = 15.22\%$ (conservative recall) of the ground truth and $Pr = 46.32\%$ (conservative precision) of all retrieved events. The more progressive recall and precision is higher with $Re^* = 33.59\%$ and $Pr^* = 94.01\%$. The distribution of the remaining error classes shows, e.g., how many fragmenting events $F'$ (206 → 25.72%) cause the fragmentations $F$ (60 → 2.46%) in the ground truth.

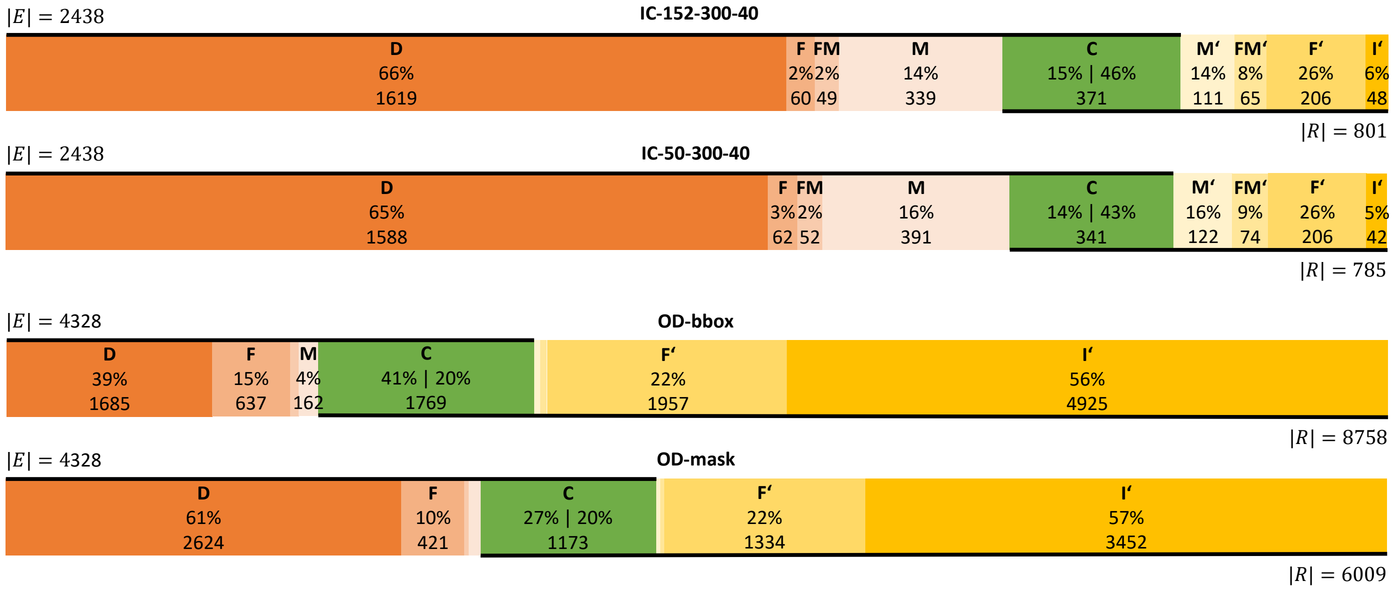
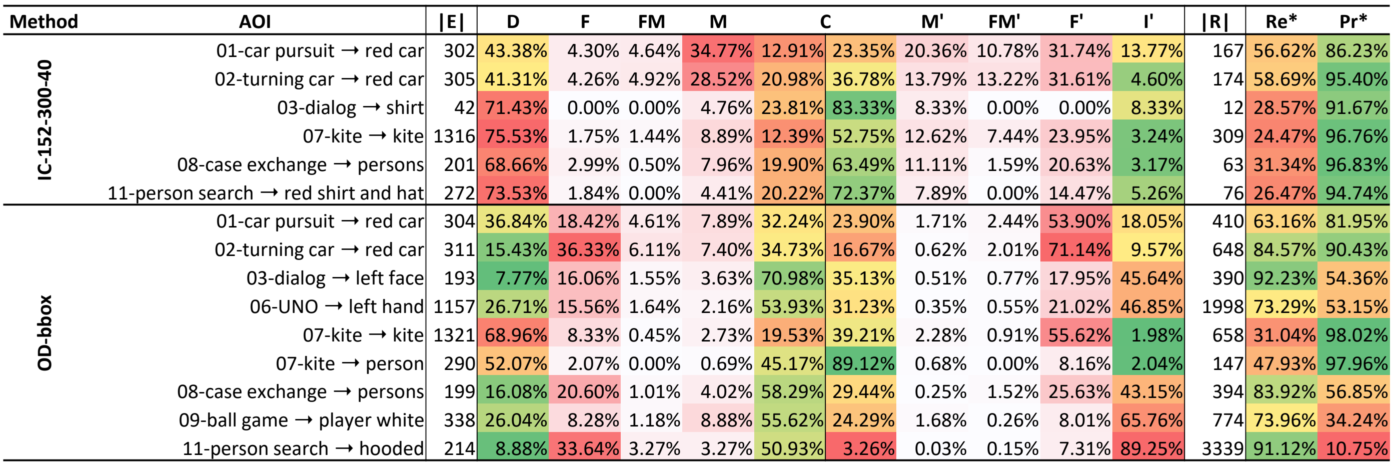


**Figure 4.** EAD diagrams visualizing the event based error classes with respect to ground truth events $|E|$ and returned events $|R|$.

The two *OD* methods have a similar distribution of error classes for retrieved events: the rate of fragmenting events is $\frac{F'}{|R|} \approx 22\%$, the rate of insertions is $\frac{I'}{|R|} \approx 56\%$, the counts for $M'$ and $FM'$ are very low, and the conservative precision $Pr$ is similar with $\frac{C}{|R|} \approx 20\%$. However, OD-bbox predicts 2749 events more than OD-mask which results in a higher absolute number of correct events for OD-bbox ($C = 1769$) compared to OD-mask ($C = 1173$). With $|E|$ being constant for both *OD* methods, $Re = \frac{C}{|E|}$ is higher for OD-bbox (40.88%) than for OD-mask (27.1%). Consequently, the fraction of deletions for OD-bbox (38.94%) is lower than the fraction for OD-mask (60.63%) which is close to the level of the *IC* methods. Further, the *OD* methods report a higher level of fragmented events $F$ than merged events $M$. We observe the opposite for *IC*. The progressive precision and recall values are $Pr^* = 43.77\%$ and $Re^* = 61.06\%$ for OD-bbox, and $Pr^* = 42.55\%$ and $Re^* = 39.37\%$ for OD-mask.

### 4.4.2. AOI Performance Breakdown

For IC-152-300-40 and OD-bbox, we report the event metrics per AOI in a table (see Figure 5). For *IC*, we observe a difference between the two "red car" AOIs and the remaining AOIs. On average, we see a lower level of deletions for "red car" with $\frac{D}{|E|} = 42.34\%$ and an increased level of merged events with $\frac{M}{|E|} = 31.65\%$, compared to the other AOIs which average to 72.29% and 6.51%, respectively. Consequently, we observe the best progressive recall for "red car" with $Re^* = 57.66\%$ on average, compared to $Re^* = 27.71\%$ for the other AOIs. The conservative precision $\frac{C}{|R|} = 30.07\%$ is lower than the average of 67.99% for the other AOIs. For "red car", $M$ is higher and $C$ is lower for `01-car pursuit` than for `02-turning car`. Hence, the conservative recall $\frac{C}{|E|} = 12.91\%$ for `01-car pursuit` is relatively lower by 38.47%, while the $Re^*$ is lower by 3.53% only. Further, we observe the highest relative number of insertions with $\frac{I'}{|R|} = 13.77\%$ (others average to 4.92%). The *OD* methods result in more diverse error class distributions. We observe a low level of $D$ in relation to ground truth events $|E|$ for "red car" (`02-turning car`), "left face", "persons", and "hooded", averaging to 12.04%. The AOIs "kite" and "person" result in the highest level for deletions $D$ with 68.96% and 52.07%, respectively. For these AOIs, the low and high levels for $D$ coincide with the highest and lowest $Re^*$ values. Overall, we see a high level of fragmented events $F$ with highest values for "hooded" with 33.64% and "red car" in `02-turning car` with 36.33%, and lowest values for "person" in `07-kite` with 2.07%. The AOI "left face" results in the best conservative recall, $Re = 70.98\%$, due to the low level of deletions $D$, with 7.77%. For insertions $I'$, we observe the lowest levels for the AOIs in `kite` with an average of 2.01%, followed by "red car" with 9.57% in `02-turning car` and 18.05% in `01-car pursuit`. All other AOIs average to 58.13% with a peak for "hooded" with 89.25%. These four AOIs with the lowest insertion rate $\frac{I'}{|R|}$ have the best progressive precision with, on average, $Pr^* = 92.09\%$. The remaining five AOIs average to $Pr^* = 41.87\%$ with the minimum for "hooded" with $Pr^* = 10.75\%$. The highest levels of fragmenting events are observed for "red car" (53.9% and 71.14%) and "kite" (55.62%). To compare the results of AOIs that remain for *IC* and *OD* methods, we calculate an event-based $f_1$ score as $f_1 = 2 \cdot \frac{Pr^* \cdot Re^*}{Pr^* + Re^*}$. For the AOIs "red car" (x2), "kite", and "persons" (`08-case exchange`), we receive $f_1$ scores of $68.36\%, 72.67\%, 39.06\%, 47.36\%$ for IC-152-300-40 and $71.34\%, 87.4\%, 47.15\%, 67.78\%$ for OD-bbox. For this selection of AOIs, OD-bbox yields the respectively better performance.



| Method | AOI | \|E\| | D | F | FM | M | C | | M' | FM' | F' | I' | \|R\| | Re* | Pr* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IC-152-300-40 | 01-car pursuit → red car | 302 | 43.38% | 4.30% | 4.64% | 34.77% | 12.91% | 23.35% | 20.36% | 10.78% | 31.74% | 13.77% | 167 | 56.62% | 86.23% |
| | 02-turning car → red car | 305 | 41.31% | 4.26% | 4.92% | 28.52% | 20.98% | 36.78% | 13.79% | 13.22% | 31.61% | 4.60% | 174 | 58.69% | 95.40% |
| | 03-dialog → shirt | 42 | 71.43% | 0.00% | 0.00% | 4.76% | 23.81% | 83.33% | 8.33% | 0.00% | 0.00% | 8.33% | 12 | 28.57% | 91.67% |
| | 07-kite → kite | 1316 | 75.53% | 1.75% | 1.44% | 8.89% | 12.39% | 52.75% | 12.62% | 7.44% | 23.95% | 3.24% | 309 | 24.47% | 96.76% |
| | 08-case exchange → persons | 201 | 68.66% | 2.99% | 0.50% | 7.96% | 19.90% | 63.49% | 11.11% | 1.59% | 20.63% | 3.17% | 63 | 31.34% | 96.83% |
| | 11-person search → red shirt and hat | 272 | 73.53% | 1.84% | 0.00% | 4.41% | 20.22% | 72.37% | 7.89% | 0.00% | 14.47% | 5.26% | 76 | 26.47% | 94.74% |
| OD-bbox | 01-car pursuit → red car | 304 | 36.84% | 18.42% | 4.61% | 7.89% | 32.24% | 23.90% | 1.71% | 2.44% | 53.90% | 18.05% | 410 | 63.16% | 81.95% |
| | 02-turning car → red car | 311 | 15.43% | 36.33% | 6.11% | 7.40% | 34.73% | 16.67% | 0.62% | 2.01% | 71.14% | 9.57% | 648 | 84.57% | 90.43% |
| | 03-dialog → left face | 193 | 7.77% | 16.06% | 1.55% | 3.63% | 70.98% | 35.13% | 0.51% | 0.77% | 17.95% | 45.64% | 390 | 92.23% | 54.36% |
| | 06-UNO → left hand | 1157 | 26.71% | 15.56% | 1.64% | 2.16% | 53.93% | 31.23% | 0.35% | 0.55% | 21.02% | 46.85% | 1998 | 73.29% | 53.15% |
| | 07-kite → kite | 1321 | 68.96% | 8.33% | 0.45% | 2.73% | 19.53% | 39.21% | 2.28% | 0.91% | 55.62% | 1.98% | 658 | 31.04% | 98.02% |
| | 07-kite → person | 290 | 52.07% | 2.07% | 0.00% | 0.69% | 45.17% | 89.12% | 0.68% | 0.00% | 8.16% | 2.04% | 147 | 47.93% | 97.96% |
| | 08-case exchange → persons | 199 | 16.08% | 20.60% | 1.01% | 4.02% | 58.29% | 29.44% | 0.25% | 1.52% | 25.63% | 43.15% | 394 | 83.92% | 56.85% |
| | 09-ball game → player white | 338 | 26.04% | 8.28% | 1.18% | 8.88% | 55.62% | 24.29% | 1.68% | 0.26% | 8.01% | 65.76% | 774 | 73.96% | 34.24% |
| | 11-person search → hooded | 214 | 8.88% | 33.64% | 3.27% | 3.27% | 50.93% | 3.26% | 0.03% | 0.15% | 7.31% | 89.25% | 3339 | 91.12% | 10.75% |

**Figure 5.** EAD table for AOIs with a non-zero recall ($dr < 90\%$; $\frac{D}{|E|} \ll 100\%$) for IC-152-300-40 and OD-bbox.

### 4.4.3. Impact of IC Parameters on Performance

The *IC* method offers multiple parameters for tuning the outcome, besides the classification *model*. We investigate the impact of $T_{dur}$ & $T_{noise}$ and $T_p$ on the frame and event metrics. With varying $T_p$, we observe no changes in the distribution of event error classes (see Figure 6). In addition, $Pr^*$ ranges between 32.69% and 33.76%, and $Re^*$ ranges between 93.58% and 94.39% for the different settings of $T_p$. When increasing the duration and noise thresholds, we observe a monotonic increase in the number of deletions $D$: the ratio $\frac{D}{|E|}$ ranges from 59.49% for 100 ms to 70.87% for 700 ms. At the same time, $\frac{M}{|E|}$ increases from 7.23% to 18.1% and $Re = \frac{C}{|E|}$ decreases from 21.89% to 6.8%. Concerning the error classes of retrieved events, we observe a monotonic decrease in the number of insertions $I'$ ranging from 10.64% for 100 ms to 3.18% for 700 ms. Similarly, $\frac{F'}{|R|}$ decreases from 44.99% to 11.74%, as well as the absolute number of retrieved events $|R|$ which ranges from 1447 to 409. The level of merging events $M'$ increases from 3.73% to 36.19% which corresponds to the increase of merged events $M$. In addition, we see a trend in the progressive precision and recall values: with increasing duration and noise threshold, $Re^*$ decreases from 40.51% to 26.13% and $Pr^*$ increases from 89.36% to 96.82%. The highest $f_1$ score of 55.74% is reached for 100 ms.

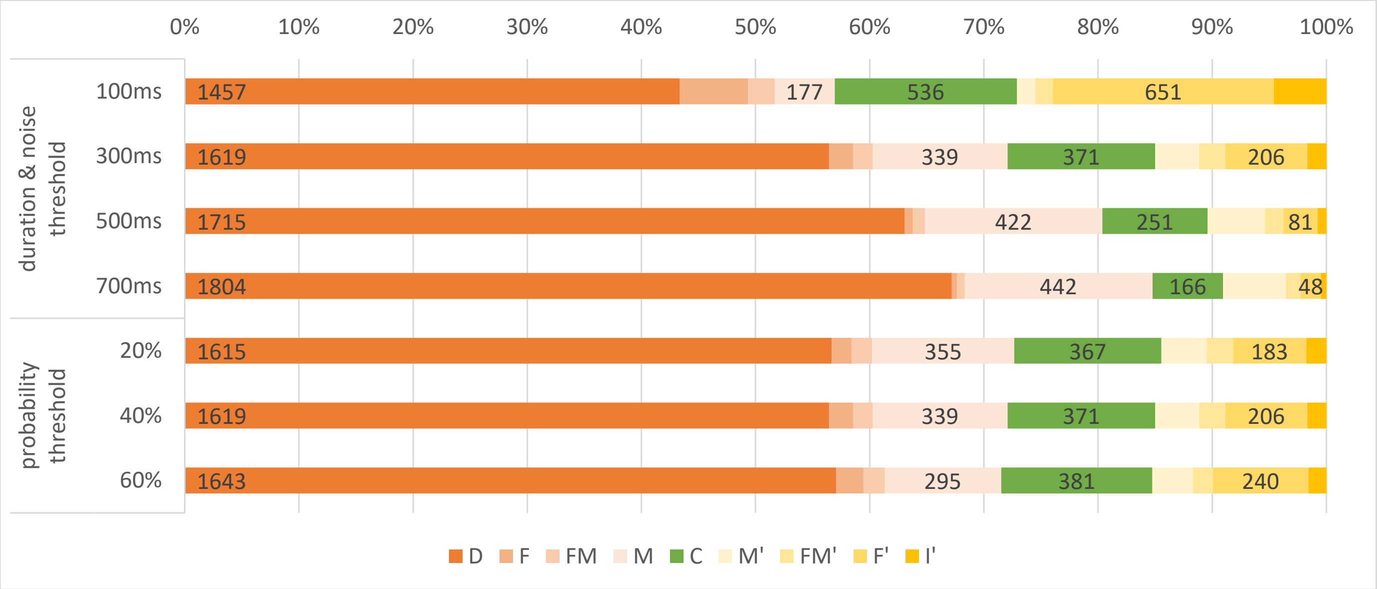


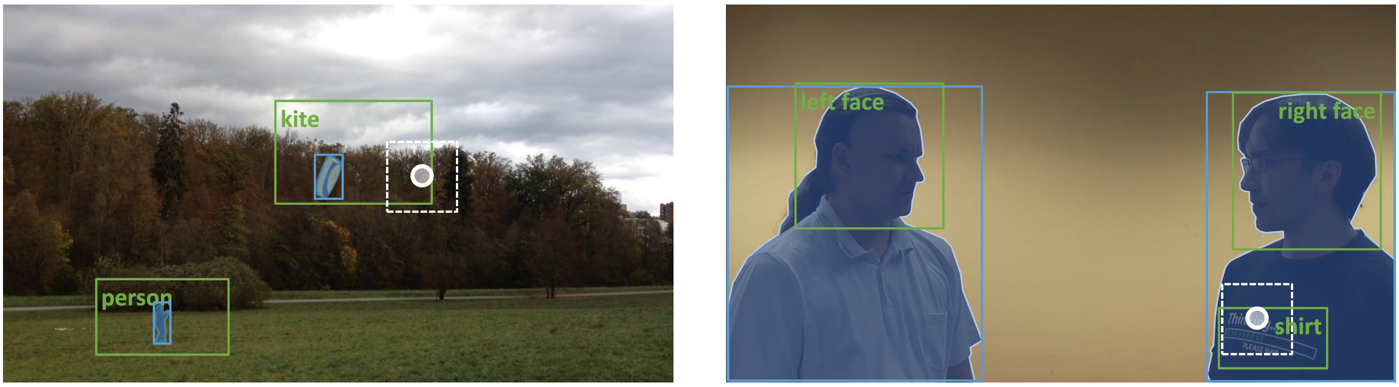
**Figure 6.** Simplified EAD diagrams for *IC* methods with varying parameters aggregated over AOIs with non-zero recall. We vary the probability threshold $T_p$ or the duration and noise threshold $T_{dur} = T_{noise}$. We use default values for other parameters. The results for 300 ms and 40% refer to the default setting IC-152-300-40 as shown in Figure 4.

## 5. Discussion

Our results show that using our methods with default parameters and the AOI configuration from Table 1 does not support all AOIs. In particular, our observations confirm that they fail in detecting visual attention for AOIs without a mapping. This affects 15 AOIs (44.12%) for *IC* and 17 AOIs (50%) for *OD*. Our results reveal 13 additional AOIs for *IC* and 8 AOIs for *OD* with weak matches that result in zero or close to zero recalls ($dr \geq 90\%$). Effectively, we count 28 AOIs (82.35%) for *IC* and 25 AOIs (73.53%) for *OD* as failing. We attribute these fails to challenge I, because the concepts of the AOIs have no adequate match to any class label of the underlying computer vision model. And, if there is a matching class label, the instances might differ from what the model has learned, i.e., from the training samples.

### *5.1. Overall Performance*

The frame and event metrics for the remaining AOIs show that deletions are the most frequent false negative error across all methods. Overall, the frame-based deletion rates *dr* are lower than the respective level of deletion events *D*. For instance, in IC-152-300-40, $\frac{D}{|E|} = 66.41\%$ of the ground truth events correspond to $dr = 36.79\%$ of the positive ground truth frames. This may indicate that our methods delete more short events than long ones. The high level of deleted events might be caused by false negatives from the computer vision model which relates to challenge I. Another problem could be that our models fail in mapping the gaze signal although the prediction was correct. To investigate this issue further, we generated videos showing the manual annotations, the gaze and fixation events, and our prediction output. We noticed that the eye tracking signal frequently suffers from low accuracy and, hence, the gaze point does not hit an AOI object even though it is obvious that the participant followed that object, e.g., a "kite". The manual annotations (bounding boxes) in the VISUS dataset are bloated up to include such erroneous gaze signals which better captures the human behavior than exact annotations. However, under the assumption that the gaze signal was accurate, this style of data annotation results in a lot of false positive ground truth events (see Figure 7a). Our methods are not robust against such cases, because they rely on local image classifications (*IC*) or fixation to object mask mapping (*OD*). Consequently, our methods report no attention events which might be one of the major reasons for the high level of deletions. We investigate this issue in detail in Section 5.4. This could also explain the difference between OD-bbox, using bounding boxes, and OD-mask, using exact object masks, i.e., OD-bbox better resamples the manual annotations and, to some degree, compensates the inaccurate gaze signals. Overall, OD-bbox shows the best progressive recall with $Re^* = 61.06\%$, while the other methods average around 35.91%. Also, our results show that *OD* yields more insertion errors than *IC* in terms of frame and event metrics: the insertion rate is $\frac{I'}{|R|} \approx 56\%$ for *OD* methods and 6% for *IC* methods. Consequently, with an average of $Pr^* = 94.33\%$, *IC* results in the better progressive precision than *OD* with $Pr^* = 43.16\%$. This suggests that the *IC* method may be the better choice for use cases with a good object to class label match, and if false negative errors are not severe. In addition, the relation of $FM, F, F'$ to $fr$ and $FM', M, M'$ to $mr$ can reveal more about the error characteristics. For instance, if we see many event errors and a low ratio of corresponding frame errors, the fragmenting or merging predictions approximate the ground truth well (see Figure 2). For instance for IC-152-300-40, merge errors $M$ make up 14% of the event errors with respect to the ground truth, but result in a low frame error rate of $mr = 0.89\%$.



(**a**) `07-kite`

(**b**) `03-dialog`

**Figure 7.** Example frames from two scenarios of the VISUS dataset [2] showing the recent fixation (white circle), ground truth annotations (green), object masks and bounding boxes for *OD* (blue), and the cropping area for *IC* (white rectangle).

### *5.2. Performance per AOI*

The results for default parameters at the AOI level show that OD-bbox performs best for the four overlapping AOIs (see Figure 5). However, all other AOIs for OD-bbox suffer from high insertion levels of more than 40%. A reason might be that these AOIs match to “person” (see Table 1) and, at least, a second AOI shares this concept, which relates to challenge II. For instance, we map the MS COCO class label “person” to the AOI “left face” in `03-dialog`, but “person” would also fit “right face” and “shirt”. The generated debug videos show that both *OD* methods detect attention events for “right face” and “shirt” based on the “person” class label. However, these are wrongly mapped to “left face” which results in a high number of false positives (see Figure 7b). This problem of the remaining AOIs is likely to cause the high level of insertion errors and the low progressive precision for *OD*, overall.

### *5.3. Impact of IC Parameters*

Our investigation with different parameters for *IC* reveals that $T_p$ is likely to have no impact on event metrics. Our assumption is that the subsequent aggregation of image classification results is a harder criterion than a high $T_p$. E.g., an incorrect classification with low probability might be dropped anyway due to reaching $T_{noise}$, because it alternates with other wrong classifications. The parameters $T_{dur}$ and $T_{noise}$ have a clear impact on the performance: increasing the threshold results in decreasing values of $Re$ and $Re^*$. $T_{dur} = T_{noise} = 100$ ms yields the best overall performance by means of the $f_1$ score, followed by the default setting which results in a better $Pr^*$, but worse $Re^*$.

### 5.4. Impact of Re-Annotating the Ground Truth Data on Deletions

In many cases, the gaze recordings from the VISUS dataset suffer from a low spatial accuracy, which resulted in coarse manual annotations. For instance in Figure 7a, the manual annotations for "person" and "kite" (green bounding boxes) are much larger than the actual object to catch the point of gaze that, when looking at the video, obviously follows the kite. In contrast, the bounding boxes and exact object masks generated by Mask R-CNN (blue rectangles and polygons) frame the "person" and the "kite" closely. Our hypothesis is, that this kind of annotation is responsible for a large portion of deletion errors (false negative events), because the ground truth reports a false attention event that cannot be captured by our detection methods. To verify our assumption, we re-annotate AOIs without a close to zero recall (see Figure 5) and repeat our analysis using the new ground truth annotations, but the same event predictions from IC-152-300-40 and OD-bbox that we have gathered in our main experiment. The videos are annotated by a single annotator and reviewed by an eye tracking expert using the Computer Vision Annotation Tool CVAT (https://github.com/openvinotoolkit/cvat, accessed on 15 June 2021). We use the polygon-based annotation feature: a polygon is created that closely frames an object at keyframes with interpolation for intermediate frames. The results show that the ratio of deletion events $\frac{D}{|E|}$ decreases by 16.3% to 50.11% for the *IC* method and by 10.3% to 28.64% for the *OD* method. Consequently, the progressive recall values $Re^*$ increase by the same amounts to 49.89% for *IC* and to 71.36% for *OD*. Thus, we can confirm our hypothesis that coarse AOI annotations increase the level of deletions. This emphasizes the importance of accurate gaze estimation methods to avoid such errors. Further, it raises the need for error-aware gaze-to-object mapping methods to compensate the impact of the gaze estimation error, similar to those presented in Barz et al. [5]. For instance, we could detect an AOI hit by checking whether the distance of a fixation point to the boundary of an AOI is smaller than a defined threshold.

### 5.5. Limitations & Future Work

Our evaluation revealed several limitations that relate to the challenges that we identified in Section 4.1.1 or to accuracy issues with the gaze signal in the VISUS dataset. The main limitation of our methods is related to challenge I: many AOIs are not supported because the concepts are not included with the pre-trained computer vision models. A promising solution to address it is to collect new samples for unsupported AOIs and AOIs with weak matches for fine-tuning the computer vision models [75,76]. We want to investigate the effectiveness of interactive machine learning methods for this purpose [77,78] compared to randomly annotating a small portion of the data as suggested in Wolf et al. [16]. Training a model from scratch, as suggested in [11,12], is not an option with state-of-the-art computer vision models, because they need a large quantity of training samples. Further, our methods offer no solution for challenge II: AOIs share the same concept. This could be solved using similarity models with interactive training. For instance, we could iteratively train a model to differentiate between "left face" and "right face" which would reduce the number of insertion errors for `03-dialog`. Using multiple object tracking algorithms [57] with humans-in-the-loop is a promising approach to support challenges III and IV. Further, we plan to develop error-aware gaze to AOI mapping similar to [5,79] to compensate for the gaze estimation error in mobile eye tracking. Also, it is likely that the fixation detection algorithm used in [2] has an impact on the ground truth extraction. The authors mentioned that, e.g., smooth pursuit movements are not supported well, which make up a large portion of the data. The selection of a suitable fixation detection algorithm is even more important for mobile eye tracking [74]. In addition, we recently showcased a real-time application of the *IC* method in an augmented reality setting with objects that are well represented in the training data of the image classification model [72], similar to Machado et al. [9], based on the AR eye tracking toolkit [80]. Our method enables a stable augmentation of ambient objects via the head-mounted display.

## 6. Conclusions

In this work, we implemented two methods for detecting visual attention using pre-trained deep learning models from computer vision. In addition, we defined an evaluation framework based on the VISUS dataset by Kurzhals et al. [2] and identified four challenges for methods that map gaze to AOIs. We used a set of fine-grained metrics by Ward et al. [18] from the field of activity recognition to evaluate our visual attention to AOI mapping methods. Our methods performed well for AOIs with distinct concepts which have a strong match to the pre-trained model classes. However, several limitations impede our goal of accelerating and objectifying AOI annotation in eye tracking research. For instance, our methods drop in performance when a concept is not supported, when two instances of the same concept cannot be disambiguated, or when gaze estimation errors occur. In the discussion, we proposed ways to overcome these limitations. In particular, we suggest to use interactive machine learning for adapting our methods to new scenarios or to differentiate between instances of the same concept. Further, we proposed an approach based on multi-object tracking to cope with AOIs that have a similar appearance.

**Author Contributions:** Conceptualization, methodology, software, formal analysis, investigation, data curation, visualization, supervision, writing—original draft preparation, M.B.; writing—review and editing, D.S.; resources, project administration, funding acquisition, M.B. and D.S.; All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the German Federal Ministry of Education and Research (Bundesministerium für Bildung und Forschung; BMBF) via the project "GeAR" (Grant No. 01JD1811C).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The VISUS dataset is provided online by Kurzhals et al. [2] at https://www.visus.uni-stuttgart.de/publikationen/benchmark-eyetracking (accessed on 12 April 2021). In addition, we provide the extracted ground truth events and predicted events for each scenario and participant as supplementary material.

**Acknowledgments:** In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## Abbreviations

**General**

| | |
|---|---|
| AOI | Area of Interest |
| IC | [Method] Detect Attention using Gaze-guided Image Classification |
| OD | [Method] Detect Attention using Object Detection |
| EAD | Event Analysis Diagram |

**Event-based Error Classes and Metrics**

$|E|$ Number of ground truth events $|E| = D + F + FM + M + C$
$|R|$ Number of predicted events, or returns $|R| = M' + FM' + F' + I' + C$
$D$ Number of deletion errors (false negatives)
$I'$ Number of insertion errors (false positives)
$F, F'$ Number of fragmentation errors
$M, M'$ Number of merge errors
$FM, FM'$ Number of fragmentation and merge errors (if both occur together)
$C$ Number of correct events
$Pr$ Event-based precision $Pr = \frac{C}{|R|}$ (conservative)
$Re$ Event-based recall $Re = \frac{C}{|E|}$ (conservative)
$Pr^*$ Event-based precision $Pr^* = \frac{|R|-I'}{|R|}$ (progressive)
$Re^*$ Event-based recall $Re^* = \frac{|E|-D}{|E|}$ (progressive)

**Frame-based Error Classes and Metrics**

$P$ Total number of positive frames
$N$ Total number of negative frames
$TP$ Number of true positives (frames)
$TN$ Number of true negatives (frames)
$D_f, dr$ Number of deletion errors (frames), deletion rate
$I_f, ir$ Number of insertion errors (frames), insertion rate
$F_F, fr$ Number of fragmentation errors (frames), fragmentation rate
$M_f, mr$ Number of merge errors (frames), merging rate
$O_f, o$ Number of overfill errors (frames), ratio of overfills
$U_f, u$ Number of underfill errors (frames), ratio of underfills
$tpr$ True positive rate
$fpr$ False positive rate

## References

1. Holmqvist, K.; Andersson, R. *Eye Tracking: A Comprehensive Guide to Methods, Paradigms and Measures*; Lund Eye-Tracking Research Institute: Lund, Sweden, 2011.
2. Kurzhals, K.; Bopp, C.F.; Bässler, J.; Ebinger, F.; Weiskopf, D. 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, Paris, France, 10 November 2014; Association for Computing Machinery: New York, NY, USA, 2014; pp. 54–60. doi:10.1145/2669557.2669558.
3. Yu, L.H.; Eizenman, M. A new methodology for determining point-of-gaze in head-mounted eye tracking systems. *IEEE Trans. Biomed. Eng.* **2004**, *51*, 1765–1773, ISBN 0018-9294 VO-51, doi:10.1109/TBME.2004.831523.
4. Pfeiffer, T.; Renner, P.; Pfeiffer-Leßmann, N. EyeSee3D 2.0: Model-based real-time analysis of mobile eye-tracking in static and dynamic three-dimensional scenes. In Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications, ETRA '16, Charleston, South Carolina, 14 March 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 189–196. doi:10.1145/2857491.2857532.
5. Barz, M.; Daiber, F.; Sonntag, D.; Bulling, A. Error-Aware Gaze-Based Interfaces for Robust Mobile Gaze Interaction. In Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications, Warsaw, Poland, 14 June 2018; ACM: New York, NY, USA, 2018; pp. 24:1–24:10. doi:10.1145/3204493.3204536.
6. Mehlmann, G.; Häring, M.; Janowski, K.; Baur, T.; Gebhard, P.; André, E. Exploring a Model of Gaze for Grounding in Multimodal HRI. In Proceedings of the 16th International Conference on Multimodal Interaction-ICMI '14, Istanbul, Turkey, 12 November 2014; ACM: New York, NY, USA, 2014; pp. 247–254. doi:10.1145/2663204.2663275.
7. Barz, M.; Sonntag, D. Gaze-guided object classification using deep neural networks for attention-based computing. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct-UbiComp '16, Heidelberg, Germany, 12 September 2016; ACM Press: New York, NY, USA, 2016; pp. 253–256. doi:10.1145/2968219.2971389.
8. Kurzhals, K.; Hlawatsch, M.; Seeger, C.; Weiskopf, D. Visual Analytics for Mobile Eye Tracking. *IEEE Trans. Vis. Comput. Graph.* **2017**, *23*, 301–310. doi:10.1109/TVCG.2016.2598695.
9. Machado, E.; Carrillo, I.; Chen, L. Visual Attention-Based Object Detection in Cluttered Environments. In Proceedings of the 2019 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), Leicester, UK, 19-23 August 2019; pp. 133–139. doi:10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00064.

10. Pontillo, D.F.; Kinsman, T.B.; Pelz, J.B. SemantiCode: Using content similarity and database-driven matching to code wearable eyetracker gaze data. In *Eye Tracking Research and Applications Symposium (ETRA)*; ACM Press: New York, NY, USA, 2010; pp. 267–270. doi:10.1145/1743666.1743729.
11. Toyama, T.; Kieninger, T.; Shafait, F.; Dengel, A. Gaze guided object recognition using a head-mounted eye tracker. In Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA '12, Santa Barbara, California, 28 March 2012; ACM: New York, NY, USA, 2012; pp. 91–98. doi:10.1145/2168556.2168570.
12. Panetta, K.; Wan, Q.; Kaszowska, A.; Taylor, H.A.; Agaian, S. Software Architecture for Automating Cognitive Science Eye-Tracking Data Analysis and Object Annotation. *IEEE Trans. Hum. Mach. Syst.* **2019**, *49*, 268–277. doi:10.1109/THMS.2019.2892919.
13. Venuprasad, P.; Xu, L.; Huang, E.; Gilman, A.; Chukoskie, L.; Cosman, P. Analyzing Gaze Behavior Using Object Detection and Unsupervised Clustering. In Proceedings of the ACM Symposium on Eye Tracking Research and Applications, Stuttgart, Germany, 02 June 2012; Association for Computing Machinery: New York, NY, USA, 2020; p. 9. doi:10.1145/3379155.3391316.
14. Brône, G.; Oben, B.; Goedemé, T. Towards a more effective method for analyzing mobile eye-tracking data: Integrating gaze data with object recognition algorithms. In Proceedings of the 1st International Workshop on Pervasive Eye Tracking & Mobile Eye-Based Interaction, PETMEI '11, Beijing, China, 18 September 2011; Association for Computing Machinery: New York, NY, USA, 2011; pp. 53–56. doi:10.1145/2029956.2029971.
15. De Beugher, S.; Brône, G.; Goedemé, T. Automatic analysis of in-the-wild mobile eye-tracking experiments using object, face and person detection. In Proceedings of the 2014 International Conference on Computer Vision Theory and Applications (VISAPP), Lisbon, Portugal, 5-8 January 2014; Volume 1, pp. 625–633.
16. Wolf, J.; Hess, S.; Bachmann, D.; Lohmeyer, Q.; Meboldt, M. Automating areas of interest analysis in mobile eye tracking experiments based on machine learning. *J. Eye Mov. Res.* **2018**, *11*, 6. doi:10.3929/ethz-b-000309840.
17. Huang, C.M.; Mutlu, B. Anticipatory robot control for efficient human-robot collaboration. In Proceedings of the 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Christchurch, New Zealand, 7-10 March 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 83–90. doi:10.1109/HRI.2016.7451737.
18. Ward, J.A.; Lukowicz, P.; Gellersen, H.W. Performance metrics for activity recognition. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–23. doi:10.1145/1889681.1889687.
19. Borji, A.; Itti, L. State-of-the-Art in Visual Attention Modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 185–207. doi:10.1109/TPAMI.2012.89.
20. Yarbus, A.L. Eye movements and vision. *Neuropsychologia* **1967**, *6*, 222. doi:10.1016/0028-3932(68)90012-2.
21. Land, M.F.; Hayhoe, M. In what ways do eye movements contribute to everyday activities? *Vis. Res.* **2001**, *41*, 3559–3565. doi:10.1016/S0042-6989(01)00102-X.
22. DeAngelus, M.; Pelz, J.B. Top-down control of eye movements: Yarbus revisited. *Vis. Cogn.* **2009**, *17*, 790–811. doi:10.1080/13506280902793843.
23. Rothkopf, C.A.; Ballard, D.H.; Hayhoe, M.M. Task and context determine where you look. *J. Vis.* **2016**, *7*, 16. doi:10.1167/7.14.16.
24. André, E.; Chai, J.Y. Introduction to the special issue on eye gaze in intelligent human-machine interaction. *ACM Trans. Interact. Intell. Syst.* **2012**, *1*, 1–3. doi:10.1145/2070719.2070720.
25. André, E.; Chai, J. Introduction to the special section on eye gaze and conversation. *ACM Trans. Interact. Intell. Syst.* **2013**, *3*, 1–2. doi:10.1145/2499474.2499479.
26. Nakano, Y.I.; Bednarik, R.; Huang, H.H.; Jokinen, K. Introduction to the special issue on new directions in eye gaze for interactive intelligent systems. *ACM Trans. Interact. Intell. Syst.* **2016**, *6*, 1–3. doi:10.1145/2893485.
27. Qvarfordt, P. Gaze-informed multimodal interaction. In *The Handbook of Multimodal-Multisensor Interfaces: Foundations, User Modeling, and Common Modality Combinations*; ACM: New York, NY, USA, 2017; Volume 1, pp. 365–402. doi:10.1145/3015783.3015794.
28. Ishii, R.; Nakano, Y.I.; Nishida, T. Gaze awareness in conversational agents: Estimating a user's conversational engagement from eye gaze. *ACM Trans. Interact. Intell. Syst.* **2013**, *3*, 11:1–11:25. doi:10.1145/2499474.2499480.
29. Ishii, R.; Otsuka, K.; Kumano, S.; Yamato, J. Prediction of Who Will Be the Next Speaker and When Using Gaze Behavior in Multiparty Meetings. *ACM Trans. Interact. Intell. Syst.* **2016**, *6*, 4:1–4:31. doi:10.1145/2757284.
30. Jokinen, K.; Furukawa, H.; Nishida, M.; Yamamoto, S. Gaze and turn-taking behavior in casual conversational interactions. *ACM Trans. Interact. Intell. Syst.* **2013**, *3*, 1–30. doi:10.1145/2499474.2499481.
31. Prasov, Z.; Chai, J.Y. What's in a Gaze?: The Role of Eye-gaze in Reference Resolution in Multimodal Conversational Interfaces. In Proceedings of the 13th International Conference on Intelligent User Interfaces, Gran Canaria, Spain, 13 January 2008; ACM: New York, NY, USA, 2008; pp. 20–29. doi:10.1145/1378773.1378777.
32. Xu, T.; Zhang, H.; Yu, C. See you see me: The role of Eye contact in multimodal human-robot interaction. *ACM Trans. Interact. Intell. Syst.* **2016**, *6*, 1–22. doi:10.1145/2882970.
33. Baur, T.; Mehlmann, G.; Damian, I.; Lingenfelser, F.; Wagner, J.; Lugrin, B.; André, E.; Gebhard, P. Context-aware automated analysis and annotation of social human-agent interactions. *ACM Trans. Interact. Intell. Syst.* **2015**, *5*, 1–33. doi:10.1145/2764921.
34. Thomason, J.; Sinapov, J.; Svetlik, M.; Stone, P.; Mooney, R.J. Learning Multi-modal Grounded Linguistic Semantics by Playing "I Spy". In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, New York, USA, 09 July 2016; pp. 3477–3483.
35. Buscher, G.; Dengel, A.; Biedert, R.; van Elst, L. Attentive documents: Eye tracking as implicit feedback for information retrieval and beyond. *ACM Trans. Interact. Intell. Syst.* **2012**, *1*, 1–30. doi:10.1145/2070719.2070722.

36. Sattar, H.; Müller, S.; Fritz, M.; Bulling, A. Prediction of search targets from fixations in open-world settings. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7-12 June 2015; IEEE Computer Society: Los Alamitos, CA, USA, 2015; pp. 981–990. doi:10.1109/CVPR.2015.7298700.
37. Sattar, H.; Bulling, A.; Fritz, M. Predicting the Category and Attributes of Visual Search Targets Using Deep Gaze Pooling. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshop (ICCVW), Venice, Italy, 22-29 October 2017; IEEE Computer Society: Los Alamitos, CA, USA, 2017; pp. 2740–2748. doi:10.1109/ICCVW.2017.322.
38. Sattar, H.; Fritz, M.; Bulling, A. Deep gaze pooling: Inferring and visually decoding search intents from human gaze fixations. *Neurocomputing* **2020**, *387*, 369–382. doi:10.1016/j.neucom.2020.01.028.
39. Stauden, S.; Barz, M.; Sonntag, D. Visual Search Target Inference using Bag of Deep Visual Words. In *KI 2018: Advances in Artificial Intelligence*; Trollmann, F., Turhan, A.Y., Eds.; Springer: Cham, Switzerland, 2018; pp. 297–304. doi:10.1007/978-3-030-00111-7_25.
40. Barz, M.; Stauden, S.; Sonntag, D. Visual Search Target Inference in Natural Interaction Settings with Machine Learning. In Proceedings of the 2020 ACM Symposium on Eye Tracking Research & Applications, Stuttgart, Germany, 02 June 2012; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1–8. doi:10.1145/3379155.3391314.
41. Bulling, A.; Weichel, C.; Gellersen, H. EyeContext: Recognition of High-level Contextual Cues from Human Visual Behaviour. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Paris, France, 27 April 2013; Association for Computing Machinery: New York, NY, USA, 2013; pp. 305–308. doi:10.1145/2470654.2470697.
42. Steil, J.; Bulling, A. Discovery of everyday human activities from long-term visual behaviour using topic models. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing-UbiComp '15, Osaka, Japan, 07 September 2015; ACM Press: New York, NY, USA, 2015; pp. 75–85. doi:10.1145/2750858.2807520.
43. Steil, J.; Müller, P.; Sugano, Y.; Bulling, A. Forecasting user attention during everyday mobile interactions using device-integrated and wearable sensors. In Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '18, Barcelona, Spain, 03 September 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 1–13. doi:10.1145/3229434.3229439.
44. Fathi, A.; Li, Y.; Rehg, J.M. Learning to Recognize Daily Actions Using Gaze. In *Computer Vision–ECCV 2012*; Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 314–327. doi:10.1007/978-3-642-33718-5_23.
45. Li, Y.; Zhefan Ye, Z.; Rehg, J.M. Delving into egocentric actions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7-12 June 2015; pp. 287–295. doi:10.1109/CVPR.2015.7298625.
46. Li, Y.; Liu, M.; Rehg, J.M. In the Eye of Beholder: Joint Learning of Gaze and Actions in First Person Video. In Proceedings of the The European Conference on Computer Vision (ECCV), Munich, Germany, 8-14 September 2018; pp. 619–635.
47. Shiga, Y.; Toyama, T.; Utsumi, Y.; Kise, K.; Dengel, A. Daily activity recognition combining gaze motion and visual features. In Proceedings of the UbiComp 2014-Adjunct Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing; Association for Computing Machinery, Inc.: New York, New York, USA, 2014; pp. 1103–1111. doi:10.1145/2638728.2641691.
48. Ramirez-Amaro, K.; Minhas, H.N.; Zehetleitner, M.; Beetz, M.; Cheng, G. Added value of gaze-exploiting semantic representation to allow robots inferring human behaviors. *ACM Trans. Interact. Intell. Syst.* **2017**, *7*, 1–30. doi:10.1145/2939381.
49. Kurzhals, K.; Rodrigues, N.; Koch, M.; Stoll, M.; Bruhn, A.; Bulling, A.; Weiskopf, D. Visual analytics and annotation of pervasive eye tracking video. In Proceedings of the Eye Tracking Research and Applications Symposium (ETRA), Stuttgart, Germany, 02 June 2012; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1–9. doi:10.1145/3379155.3391326.
50. Steichen, B.; Conati, C.; Carenini, G. Inferring visualization task properties, user performance, and user cognitive abilities from eye gaze data. *ACM Trans. Interact. Intell. Syst.* **2014**, *4*, 1–29. doi:10.1145/2633043.
51. Conati, C.; Lallé, S.; Rahman, M.A.; Toker, D. Comparing and Combining Interaction Data and Eye-tracking Data for the Real-time Prediction of User Cognitive Abilities in Visualization Tasks. *ACM Trans. Interact. Intell. Syst.* **2020**, *10*, 12. doi:10.1145/3301400.
52. De Beugher, S.; Ichiche, Y.; Brône, G.; Goedemé, T. Automatic analysis of eye-tracking data using object detection algorithms. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp '12, Pittsburgh, Pennsylvania, 05 September 2012; Association for Computing Machinery: New York, NY, USA, 2012; pp. 677–680. doi:10.1145/2370216.2370363.
53. Evans, K.M.; Jacobs, R.A.; Tarduno, J.A.; Pelz, J.B. Collecting and Analyzing Eye-Tracking Data in Outdoor Environments. *J. Eye Mov. Res.* **2012**, *5*, 19. doi:10.16910/jemr.5.2.6.
54. Fong, A.; Hoffman, D.; Ratwani, R.M. Making Sense of Mobile Eye-Tracking Data in the Real-World: A Human-in-the-Loop Analysis Approach. *Proc. Hum. Factors Ergon. Soc. Annu. Meet.* **2016**, *60*, 1569–1573. doi:10.1177/1541931213601362.
55. Panetta, K.; Wan, Q.; Rajeev, S.; Kaszowska, A.; Gardony, A.L.; Naranjo, K.; Taylor, H.A.; Agaian, S. ISeeColor: Method for Advanced Visual Analytics of Eye Tracking Data. *IEEE Access* **2020**, *8*, 52278–52287. doi:10.1109/ACCESS.2020.2980901.
56. Sümer, O.; Goldberg, P.; Stürmer, K.; Seidel, T.; Gerjets, P.; Trautwein, U.; Kasneci, E. Teachers' Perception in the Classroom. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Salt Lake City, Utah, 19-21 June 2018; pp. 2315–2324.
57. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. SiamRPN++: Evolution of Siamese Visual Tracking With Very Deep Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15-20 June 2019; pp. 4282–4291.

58. Cao, Q.; Shen, L.; Xie, W.; Parkhi, O.M.; Zisserman, A. VGGFace2: A Dataset for Recognising Faces across Pose and Age. In Proceedings of the 2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018), Xi'an, China, 15-19 May 2018; pp. 67–74. doi:10.1109/FG.2018.00020.
59. Callemein, T.; Van Beeck, K.; Brône, G.; Goedemé, T. Automated Analysis of Eye-Tracker-Based Human-Human Interaction Studies. In *Information Science and Applications 2018*; Lecture Notes in Electrical Engineering; Kim, K.J., Baek, N., Eds.; Springer: Singapore, 2019; pp. 499–509. doi:10.1007/978-981-13-1056-0_50.
60. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110.
61. Toyama, T.; Sonntag, D. Towards episodic memory support for dementia patients by recognizing objects, faces and text in eye gaze. In *KI 2015: Advances in Artificial Intelligence*; Springer: Cham, Switzerland, 2015; Volume 9324, pp. 316–323. doi:10.1007/978-3-319-24489-1.
62. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper With Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7-12 June 2015.
63. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A.C.; Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis. (IJCV)* **2015**, *115*, 211–252. doi:10.1007/s11263-015-0816-y.
64. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22-29 October 2017; pp. 2961–2969.
65. Batliner, M.; Hess, S.; Ehrlich-Adám, C.; Lohmeyer, Q.; Meboldt, M. Automated areas of interest analysis for usability studies of tangible screen-based user interfaces using mobile eye tracking. *AI EDAM* **2020**, *34*, 505–514. doi:10.1017/S0890060420000372.
66. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *CoRR* **2017**, arXiv:1704.04861.
67. Lin, T.Y.; Maire, M.; Belongie, S.J.; Bourdev, L.D.; Girshick, R.B.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. *CoRR* **2014**, arXiv:1405.0312.
68. Zhao, Z.; Zheng, P.; Xu, S.; Wu, X. Object Detection With Deep Learning: A Review. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 3212–3232. doi:10.1109/TNNLS.2018.2876865.
69. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27-30 June 2016; pp. 770–778.
70. Ma, M.; Fan, H.; Kitani, K.M. Going Deeper into First-Person Activity Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27-30 June 2016; pp. 1894–1903.
71. Bertasius, G.; Park, H.S.; Yu, S.X.; Shi, J. First-Person Action-Object Detection with EgoNet. *CoRR* **2016**, arXiv:1603.04908.
72. Barz, M.; Kapp, S.; Kuhn, J.; Sonntag, D. Automatic Recognition and Augmentation of Attended Objects in Real-time using Eye Tracking and a Head-mounted Display. In Proceedings of the ACM Symposium on Eye Tracking Research and Applications, ETRA '21 Adjunct, Virtual Event, Germany, 25 May 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 1–4. doi:10.1145/3450341.3458766.
73. Wu, Y.; Kirillov, A.; Massa, F.; Lo, W.Y.; Girshick, R. Detectron2. 2019. Available online: https://github.com/facebookresearch/detectron2 (accessed on 2 May 2021).
74. Steil, J.; Huang, M.X.; Bulling, A. Fixation detection for head-mounted eye tracking based on visual similarity of gaze targets. In Proceedings of the Eye Tracking Research and Applications Symposium (ETRA), Warsaw, Poland, 14 June 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 1–9. doi:10.1145/3204493.3204538.
75. Käding, C.; Rodner, E.; Freytag, A.; Denzler, J. Fine-Tuning Deep Neural Networks in Continuous Learning Scenarios. In *Computer Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, 20–24 November 2016, Revised Selected Papers, Part III*; Chen, C.S., Lu, J., Ma, K.K., Eds.; Springer: Cham, Switzerland, 2017; pp. 588–605. doi:10.1007/978-3-319-54526-4_43.
76. Sonntag, D.; Barz, M.; Zacharias, J.; Stauden, S.; Rahmani, V.; Fóthi, Á.; Lörincz, A. Fine-tuning deep CNN models on specific MS COCO categories. *CoRR* **2017**, arXiv:1709.01476.
77. Simard, P.; Amershi, S.; Chickering, M.; Edelman Pelton, A.; Ghorashi, S.; Meek, C.; Ramos, G.; Suh, J.; Verwey, J.; Wang, M.; Wernsing, J. Machine Teaching: A New Paradigm for Building Machine Learning Systems. *CoRR* **2017**, arXiv:1707.06742.
78. Dudley, J.J.; Kristensson, P.O. A Review of User Interface Design for Interactive Machine Learning. *ACM Trans. Interact. Intell. Syst.* **2018**, *8*, 1–37. doi:10.1145/3185517.
79. Barz, M.; Daiber, F.; Bulling, A. Prediction of Gaze Estimation Error for Error-Aware Gaze-Based Interfaces. In Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications, ETRA '16, Charleston, South Carolina, 14 March 2016; ACM Press: New York, NY, USA, 2016; pp. 275–278. doi:10.1145/2857491.2857493.
80. Kapp, S.; Barz, M.; Mukhametov, S.; Sonntag, D.; Kuhn, J. ARETT: Augmented Reality Eye Tracking Toolkit for Head Mounted Displays. *Sensors* **2021**, *21*, 2234. doi:10.3390/s21062234.