Automatic analysis of in-the-wild mobile eye-tracking experiments using object, face and person detection

Stijn De Beugher[1], Geert Brône[2] and Toon Goedemé[1]
[1] *EAVISE, ESAT - KU Leuven , Belgium*
[2] *MIDI Research Group - KU Leuven, Belgium*
stijn.debeugher@thomasmore.be,geert.brone@arts.kuleuven.be,tgoedeme@esat.kuleuven.be

Keywords: Mobile Eye-tracking, Object Detection, Object Recognition, Real World, Data Analysis, Person Detection.

Abstract: In this paper we present a novel method for the automatic analysis of mobile eye-tracking data in natural environments. Mobile eye-trackers generate large amounts of data, making manual analysis very time-consuming. Available solutions, such as marker-based analysis minimize the manual labour but require experimental control, making real-life experiments practically unfeasible. We present a novel method for processing this mobile eye-tracking data by applying object, face and person detection algorithms. Furthermore we present a temporal smoothing technique to improve the detection rate and we trained a new detection model for occluded person and face detections. This enables the analysis to be performed on the object level rather than the traditionally used coordinate level. We present speed and accuracy results of our novel detection scheme on challenging, large-scale real-life experiments.

1 INTRODUCTION

The development of mobile eye-tracking systems has opened up the paradigm of eye-tracking to a wide variety of research disciplines and commercial applications. Whereas traditionally, the analysis of eye gaze patterns was largely confined to controlled lab-based conditions due to technological restrictions (i.c. obtrusive hardware restricting the flexibility of use and potential research questions), mobile systems allow for eye-tracking in the wild, without a necessarily predefined set of research conditions. Because of this increased flexibility, research into visual behaviour and real-life user experience now extends to natural environments such as public spaces (train stations, airports, museums, etc.), commercial environments (supermarkets, shopping centers, etc.) or to interpersonal communicative settings (helpdesk interactions, lectures, face-to-face communication, etc.). A mobile eye-tracker, as illustrated in figure 1, combines two types of cameras. The scene camera is looking forward and captures the field of view, while the eye-camera(s) on the other hand capture the eye-movements, also known as gaze data. Output of such an eye-tracker, as shown in the right part of this figure, consists of the images captured by the scene camera with the gaze-locations laid on top of them.

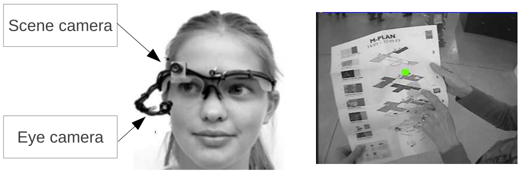

Figure 1: Left: illustration of a mobile eye-tracker consisting of a scene camera and an eye camera. Right: output of a mobile eye-tracker in which the data of the scene camera and the captured gaze point (green dot) are combined.

One of the key challenges for this new type of pervasive eye-tracking, and mobile eye-tracking in general, is the processing of data generated by the systems. By abandoning the traditional well-controlled lab-based conditions, the data stream generated by the eye-trackers becomes highly complex, both in terms of the objects and scenes that are encountered, and the gaze data that need to be analyzed and interpreted. How can researchers avoid the painstaking task of manually coding large amounts of data, which is extremely time-consuming, without losing the full potential of mobile eye-tracking systems?

Eye-tracking experiments are mostly performed in order to measure how often and for how long the test subjects looked at a specific object and/or at persons, to gather information about what 'catches the eye' in a certain setting. Recently, several solutions to the analysis problem have been proposed, some of which have been integrated in commercially available systems, see (Evans et al., 2012) for an overview. The best-known technique is the use of markers to predefine potential Areas Of Interest (AOI). These systems, which either use physical infrared markers (e.g. Tobii Glasses) or natural markers (e.g. SMI Eye Tracking Glasses), determine the boundaries of the Areas Of Analysis (AOA), generating a two-dimensional plane within which eye gaze data can be collected for longer stretches of time and generalized across subjects. The output of this type of analysis is often represented in heat maps or opacity maps that highlight the zones within the AOA that received most visual attention (measured in terms of visual fixations and fixation times). Despite their advantages in comparison to manual analysis, marker based systems suffer from a range of limitations, as discussed in (Brône et al., 2011) and (Evans et al., 2012), including the need for fixed positions of relevant objects to be tracked, and the sensitivity to the observer's position. These shortcomings impose limitations on the efficient use of (mobile) eye-tracking in real-life settings with moving subjects, objects and a dynamic environment.

This paper presents an alternative to the AOI-based methods, building on recent studies combining object recognition algorithms with eye-tracking data (De Beugher et al., 2012), (Toyama et al., 2012) and (Yun et al., 2013). By mapping gaze data on objects and object classes to be recognized in the scene video data, a number of restrictions of AOI-based approaches no longer hold, including the need to work with predefined static areas. Objects for which gaze data statistics need to be generated can be selected in the actual videostream, without prior training. The schematic representation in figure 2 shows the detection methods that were used in our approach to generate both graphical and statistical output.

The next section illustrates potentional real-life applications of our approach. In section 3 we discuss the technological background of the proposed approach, including a comparison of different potential feature extraction methods and an overview of face and body detection algorithms. In section 4 we will clarifiy our technical approach and in section 5 we report on our experiments.

2 TARGETED APPLICATIONS

In order to show the potential of our object-based eye-tracking data analysis, we detail a selection of real-world applications that can serve as both a test case and a show case for our technique: user experience, market research and (psycho)linguistics.

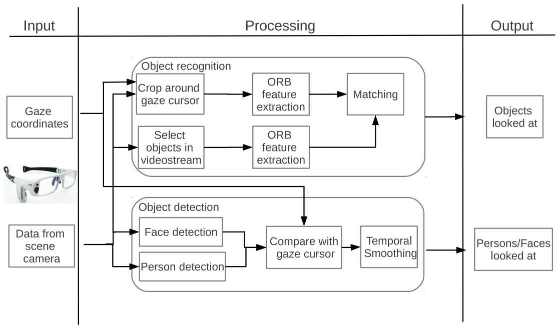

Figure 2: High-level overview of the approach. Output is generated in the form of both a visual summary of the eye-tracker data and statistical information.

2.1 Market Research

Our first case study focuses on the analysis of eye-tracker data recorded during a shopping experiment. We have chosen this case because there has been a substantial interest in eye-tracking for shopper research for several years. Indeed, these experiments yield an objective measurement of how well products catch the eye in a shop. Market researchers and (brand) developers benefit from insights into the effect of package design, shelf placement, and store planning on shopper experience. The specific context of a supermarket presents a series of challenges, for example a multitude of objects with different shapes and colors, products presented in groups on the shelves or products within the same range exhibiting similar features. As explained in the introduction, the limitations of using predefined areas of analysis makes it virtually impossible to process large-scale shopping experiments.

Our proposed object recognition method on the other hand, allows for a fast analysis of multiple objects and object categories. For instance, once the system has been trained for a specific product, it will recognize this product type each time it appears within the visual field of a test person walking through a shop and even picking up those products.

The main advantage of our system, compared to the marker-based methods, is that we no longer require predefined areas of analysis, making it possible to perform eye-tracker experiments in larger, real-world shopping environments.

2.2 Customer Journey

A prominent field of application for mobile eye-tracking is customer journey analysis. The main purpose of customer journey research of a company is gaining insights into the experience of customers. Mobile eye-trackers provide potentially useful information on customer experience, particularly when the paradigm is combined with other sensors, such as wearable EEG devices (Alves et al., 2012).

Customer experience can be measured through using so called touchpoints, the contact moments between the customer and the company e.g. in the case of advertising, communication with desk members, etc. The recordings of the mobile eye-tracker can be used to analyse the visual behaviour towards the fysical and human touchpoints. Our approach can be used to perform these analysis automatically.

In our case study we tackle the analysis of one type of customer journey involving a series of touchpoints, namely a museum visit (see section 5.3). Analysis of this data includes wayfinding analysis, analysis of human contacts, visual behaviour towards specific works of arts, etc.

2.3 Human-human Communication

A third example application tackles the analysis of eye-tracker data of a human-human communication experiment. Recent research on multimodal human-human communication has explored the role of gaze in turntaking and feedback in face-to-face conversation (Jokinen et al., 2009), shared gaze in dialogue and the function of gaze as a directive instrument in communication (Brône et al., 2010). Among the questions that are addressed in this field are: *Does a speaker visually address his/her audience during a presentation? How does the audience divide its visual attention between a speaker and relevant artefacts?* In this case, the specific challenge for the object-based eye-tracking system resides in the recognition of human bodies (and body parts), and the automatic analysis of attentional distribution between multiple objects. This test will allow for a first insight into the system's reliability for the analysis of preferential looking in communication (e.g. a speaker looking at audience vs. notes).

3 RELATED WORK

In the introduction, we already mentioned the concepts object recognition and object detection. The task of object recognition consists of retrieving a given object that is identical to a trained object in a set of images. Object detection on the other hand has extended the principle of detecting objects with a known specific appearance towards detecting objects based on a general object class model that contains intra-class variability. The next subsections describe a selection of techniques of both object recognition and object detection that can be used in our application.

3.1 Object Recognition Techniques

Object recognition, or finding an object that is identical to a trained one, is traditionally done with *local feature matching techniques*. Recognition methods define local interest regions in an image, based on specific features of the image content, which are described with descriptor vectors. The characterisation of these local regions with descriptor vectors that are invariant to changes in illumination, scale and viewpoint enables the regions to be compared across images. A survey of object recognition methods is given in (Tuytelaars and Mikolajczyk, 2008), while (Mikolajczyk et al., 2005) report comparative experiments.

Renowned techniques are the Scale Invariant Feature Transform (SIFT) (Lowe, 2004) and Speeded Up Robust Features (SURF) by (Bay et al., 2006). Although SIFT and SURF are regarded as state-of-the-art, we opted for a class of more recently developed techniques due to licensing regulations. We compared two competitive alternatives for SIFT and SURF, namely ORB (Rublee et al., 2011) and BRISK (Leutenegger et al., 2011).

The ORB feature descriptor is built on the well-known FAST keypoint detector (Rosten and Drummond, 2005) and the recently developed BRIEF descriptor (Calonder et al., 2010). ORB is a computationally efficient replacement for SIFT and SURF, it has similar matching performance and is even less affected by image noise. ORB is suitable for real-time performance since it is faster than both SURF and SIFT. Another competitive approach to keypoint detection and description is Binary Robust Invariant Scalable Keypoints (BRISK) and is as performant as the state-of-the-art algorithms, but with a significantly lower computational cost.

An evaluation of these detectors is presented by (Miksik and Mikolajczyk, 2012). Although these results demonstrate that BRISK outperforms ORB, we chose to use ORB in our algorithm based on our own experiments. Mobile eye-trackers are often equipped with low-resolution scene cameras, for example 320 by 240 px on the Arrington mobile eye-tracker. Moreover, on top of this low resolution we are only interested in a specific region around the gaze cursor, yielding a final ROI of approximately 120 by 120 px. We noted that applying BRISK to such small images often results in an insufficient number of extracted keypoints as shown in figure 3, and thus does not generate an adequate number of matches.

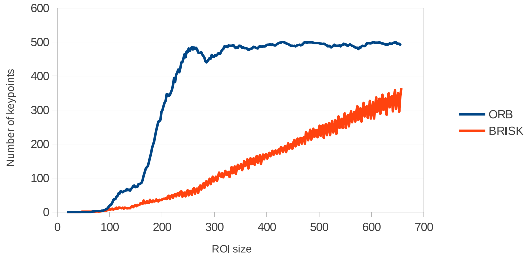

Figure 3: Comparison between ORB and Brisk. X-axis is value of both width and height of the image.

3.2 Object Detection Techniques

Several studies in the field of visual behaviour have shown that visual attention is particularly attracted to other persons (Judd et al., 2009; van Gompel, 2007) and faces (Henderson, 2003). To broaden the use of our approach we implemented techniques to automatically detect whether a person looked at another person. We make a distinction between specifically looking at a face, e.g. during talking, and looking at someone from a larger distance.

Since each body or face is unique it is impossible to apply the previously mentioned object recognition techniques. Object detection on the other hand can be described as detecting instances of objects of a certain class, in which the appearance of objects may vary, such as humans or faces, and therefore it is suitable for this purpose. A short overview of two robust object detection algorithms is described below.

The technique presented by (Viola and Jones, 2001) has proven to be a very useful tool to detect faces in natural images. This technique combines a set of weak classifiers into a final strong classifier and uses a sliding window approach to search for specific patterns in the image. Haar features models can be used to detect faces, eyes or mouths, etc. A main drawback of this technique is the limited viewing angle for which standard models can be used.

On the other hand, for full human bodies a state-of-the-art detectors are the Deformable Parts Model (DPM) (Felzenszwalb et al., 2010), Integral Channel Features (Dollár et al., 2009) or Random Hough Forests (Gall and Lempitsky, 2009). This technique uses a parts extension of HOG (Dalal and Triggs, 2005), and is therefore invariant to various postures or viewing angles. Models trained on the PASCAL and INRIA Person datasets have proven to be very robust in cases where a full body is visible, but sometimes fail when a body is not visible from head to foot (Dollár et al., 2012). Unfortunately, since the scene camera of the eye-tracker has a restricted vertical viewing angle, people not too far away appear cropped in the image, as illustrated in figure 4. In eye-tracking experiments we are often interested in the interaction between people and thus we capture lots of such images.

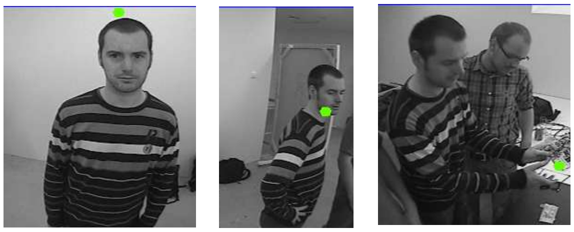

Figure 4: Example eye-tracker images in which a complete human body is not visible from head to foot.

To summarize this section we conclude that we selected ORB to detect if one looked at specific objects. To find out if one looked at another person we chose Viola and Jones and DPM based algorithms.

4 TECHNICAL APPROACH

The input of our algorithm consists of a videostream, captured by the scene camera of an eye-tracker, and a data file which contains the corresponding gaze locations. As explained in the previous chapter, we apply two different techniques to analyse the eye-tracking data. The first part of this section discusses the implementation of the ORB technique to detect how often and for how long a particular object was viewed. The second part handles the implementation of techniques to count how often and for how long a face or a person was viewed.

4.1 Recognition of Specific Objects

This part of our approach focuses on how we process eye-tracker data to generate basic statistics for specific objects to be detected. This is done in five steps:

1. Preprocessing step: since we are only interested in the objects that appear close to the visual fixation point, the input images of the forward looking camera are cropped around the gaze coordinates. Based on experiments, we chose to crop a ROI of 120 by 120 px around the gaze cursor.

2. In the next step the user selects objects of interest in the datastream by simply clicking on them while the video is playing. These objects are then stored in an object database, avoiding the tedious task of manually creating such a database with training images of the objects, as proposed in other approaches (Toyama et al., 2012; De Beugher et al., 2012).

3. The third step consists of searching for correspondences between each cropped frame and each frame stored in the database, using ORB features. We apply a matching algorithm, based on the Euclidean distance to find similar keypoints between each image pair. Furthermore we also apply several filter techniques to eliminate weak or false matches. First the distance between the two best matches is evaluated: if this distance is large enough it is safe to accept the first best match, since it is unambiguously the best choice. Secondly, a symmetrical matching scheme is used, which imposes that for a pair of matches, both points must be the best matching feature of the other. The last step involves a fundamental matrix estimation method based on RANSAC (Fischler and Bolles, 1981) to remove the outliers. This approach ensures that when we match feature points between two images, we only keep those matches that fall onto the corresponding epipolar lines.

4. In the fourth step we assign a score S to each pair of images:

$$S = \frac{\sum_{i=1}^{m} d(k_i, k_i')}{m(\sum_{i=1}^{m} A(k_i) + \sum_{i=1}^{m} A(k_i'))}, \tag{1}$$

   where k_i and k_i' stands for the ith keypoint of the corresponding images, m is the total number of matches and $A(k_i)$ stands for the size of the corresponding features. This score S is then used to decide whether a cropped frame exhibits sufficient agreement to one of the frames in the database by comparing S to a tunable threshold.

5. In the fifth and final step we cluster consecutive similar frames into a ”visual fixation”. We define a visual fixation as a series of images in which the same object was viewed with a minimal duration of 100 ms or three consecutive frames. This minimal length factor allows us to remove many false detections.

An example result of this algorithm is shown in figure 5. A total of four objects were selected from an eye-tracker experiment. In this figure we display a single image per visual fixation for each frame of interest.

4.2 Detection of Faces and Bodies

The second part of our approach focuses on the detection of faces and human bodies as it is nessecary for e.g. customer journey experiments (see section 2.2).

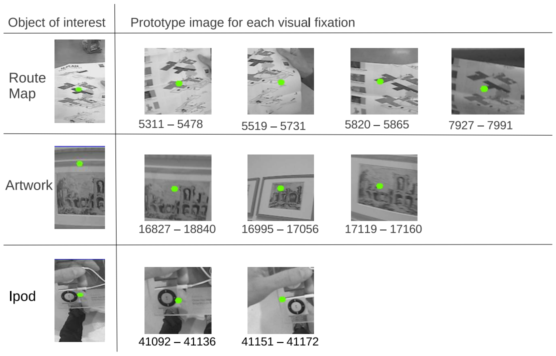

Figure 5: Results of the object detection algorithm. First and last frame number of each visual fixation is displayed.

As explained in the previous chapter, we use the standard Viola and Jones technique for face detection in combination with the standard OpenCV frontal face Haar-cascade model and the standard DPM technique (Felzenszwalb et al., 2010) in combination with a new trained 60% model, as will be explained below. We apply those techniques to the images captured by the scene camera of the eye-tracker, but we only keep the bounding boxes that are close to the gaze coordinates of the corresponding frame.

This basic implementation performs sufficiently well, but there is still room for improvement. We propose a temporal smoothing technique (see figure 6) by using the gaze-data to improve the detection rate, thus minimizing both false positives and false negatives. To reduce the number of false positives, we assume that a valid face/person detection should stand for at least a certain time (tunable via a threshold, for example 100 ms or 3 subsequent frames). This criterion substantially reduces the number of false positives (since many false detections occur occasionaly). On the other hand, if we find gaps between detection sequences, we can assume those are missing detections. Predicting them will improve the detection rate and thus further reduce the number of false negatives.

The Haar-cascade method works best for frontal faces, which is ideally suited for this application where we want to count face-to-face interactions. However if a face is presented in profile, detection will often fail. There are some models designed for

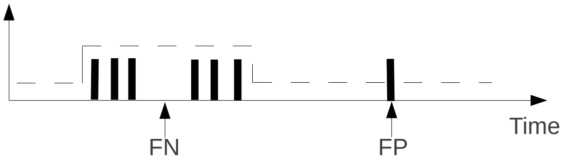

Figure 6: Vertical bars: real detections. Dashed line: output of the temporal smoothing.

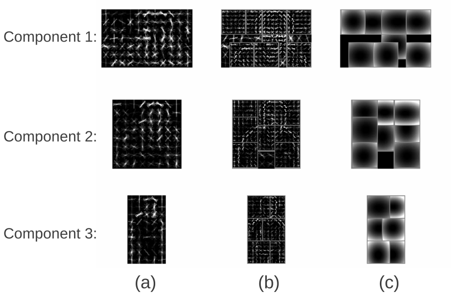

Figure 7: Components of the torso model.

profile face detection, but we saw little or no advantage with this as compared to the standard frontal face model. The DPM is very robust in the detection of full bodies, but because in our application persons are mostly not visible from head to foot (see figure 4), we used a self-trained torso model instead of the standard person model.

To overcome these shortcomings, we have trained a new model based on the standard PASCAL VOC dataset[1]. This new model is trained using only the upper 60% of the labeled bounding boxes of human bodies, resulting in a human torso model as illustrated in figure 7. Our model consists of three components, each belonging to a specific viewpoint. Every component is defined by a root filter (a), several part filters (b) and a spatial model for the location of each part relative to the root (c). This approach to cope with image border occlusion is also followed by (Mathias et al., 2013), but for a channel features detector. To the best of our knowledge, we are the first to use it on a DPM-detector. A second advantage of this cropped model is the possibility to use the first component as an upper body (head and shoulder)-detector. This model is, compared to the Haar-cascade model, robust to various poses of the head.

5 EXPERIMENTAL RESULTS

In order to test our person and object detection scheme on real-life eye-tracker experiments, we recorded a large set of eye-tracking experiments. These experiments included (i) a subject walking through a university campus building and looking for signs, (ii) a subject walking through the streets, while paying attention to traffic and other signs, (iii) a subject attending a presentation given by a lecturer and (iv) a larger experiment where multiple participants visited a museum. In this last experiment fourteen participants (7 male - 7 female) were recorded while they visited a special exhibition at Museum M in Leuven (Belgium), starting from the ticket counter all the way to the gift shop. The goal of this experiment was to determine the ease-of-use and experience of the self-guided tour: signage, information, view time of specific works, etc. Recordings were made with Tobii Glasses and Arrington Gig-E60 mobile systems and resulted in 630 minutes of video material.

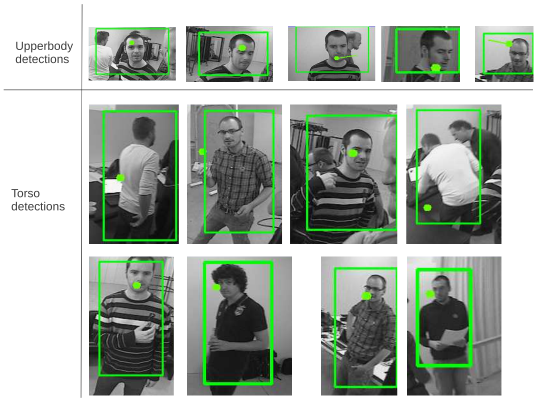

Figure 8: Example of the full body person detection.

5.1 Object Recognition Results

We tested our object recognition technique on a labeled set of images, captured during the above-mentioned experiment (i). This set consists of 2000 images with 716 labeled objects of six different categories. A precision-recall curve indicating the performance of our detector on this test set is shown in figure 9. The obtained detection results are satisfactory for most of the objects. However, a large scale variance results in a lower detection rate, as illustrated by the curve of the toilet sign, which is looked at both from very far away and from close by. Table 1 shows the execution time for a given number of selected objects of interest and a given number of video frames. As illustrated in this table, data of an eye-tracker experiment of 6000 frames (3m 20s of video data) can

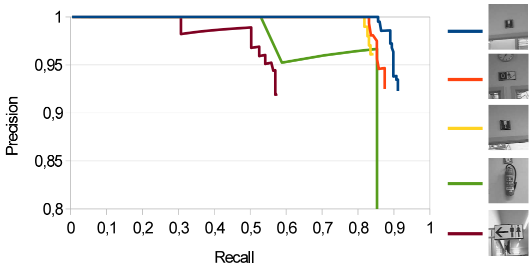

Figure 9: Precision-recall curve of our object recognition technique tested on a set of 2000 images.

[1] The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Dataset http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html

Table 1: Computational time of the object recognition implementation.

# selected objects	2	3	4	5
video of 1m 6s	31 s	42 s	54 s	68 s
video of 2m 13s	61 s	80 s	104 s	133 s
video of 3m 20s	94 s	122 s	162 s	201 s

be processed in a couple of minutes, less than the duration of the video itself. These tests were performed on a normal recent desktop PC.

5.2 Results of Face and Body Detections

In order to present results of the face and torso detections, we have labeled a set of 3000 images captured during the museum visit. In this labeling we made a distinction between looking at an upper body and looking at a person. This distinction is made in order to detect when the subject is talking to someone, or when he/she just looks at another person. In figure 10 we present a set of precision-recall curves displaying the improvements we have made. A detection is counted if it corresponds to either an upper body label or a torso label. The first curve shows the performance of the standard VOC 2009 model on our dataset. The second curve shows the performance of the standard VOC 2009 model in combination with our temporal smoothing approach. The last curve shows the new torso model in combination with our temporal smoothing. Mainly in the recall region between 0.8 and 0.9, we reached a significant improvement compared to the standard model. Indeed, with our technique, the Mean Average Precision (MAP) value increases by 6%.

As mentioned in the previous section, it is possible to use the first component of our model to detect upper bodies, and thus use this model as an alternative/additional technique to the Haar-cascade face model. An illustration of those results is presented in figure 11. It is clear that our combined Haar-cascade/upper body DPM model (yellow curve) is significantly better than both the Haar-cascade detector (blue square) or our DPM upper body model (yellow) used separately.

5.3 Combined Results of Objects, Face and Body Detections

An overview of the automatically generated output of our algorithm for the museum experiment is given in figure 12. In this figure we show a visual timeline indicating how often a visitor looked at a specific object such as a route map, a specific work of art, etc. Furthermore it indicates the interaction to other persons, for example when buying a ticket or asking information about the exhibition.

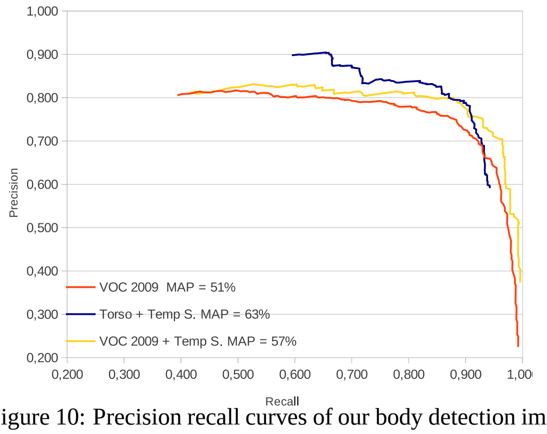

Figure 10: Precision recall curves of our body detection implementation compared to a standard model.

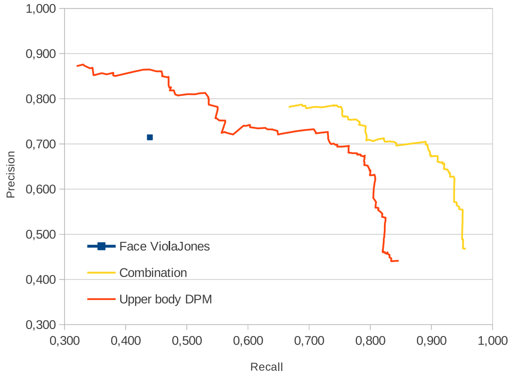

Figure 11: Precision recall curves of face detection compared to upper body detections.

6 CONCLUSION AND FUTURE WORK

In this paper we presented an approach for automatic eye-tracker data processing based on object, face and person detection. As opposed to (Toyama et al., 2012) and (De Beugher et al., 2012) we presented an object detection scheme in which a separate training is no longer required. On top of the object detection we presented an approach suited for counting how often and for how long one looked at a person or a face. In order to further improve the detection rate we proposed two novelties. The first is a temporal smoothing approach to avoid many false positives and false negatives. The second is the training of a new DPM model which is designed for torso and upper body detections. We illustrated the accuracy and performance of our approach and gave a comparison to standard techniques and presented results of large-scale real-life experiments. Our future work concentrates on intelligent sampling which should avoid the processing of every eye-tracker tick and thus reduce processing time. We will also pay attention to a more user-friendly visualisation of the results.

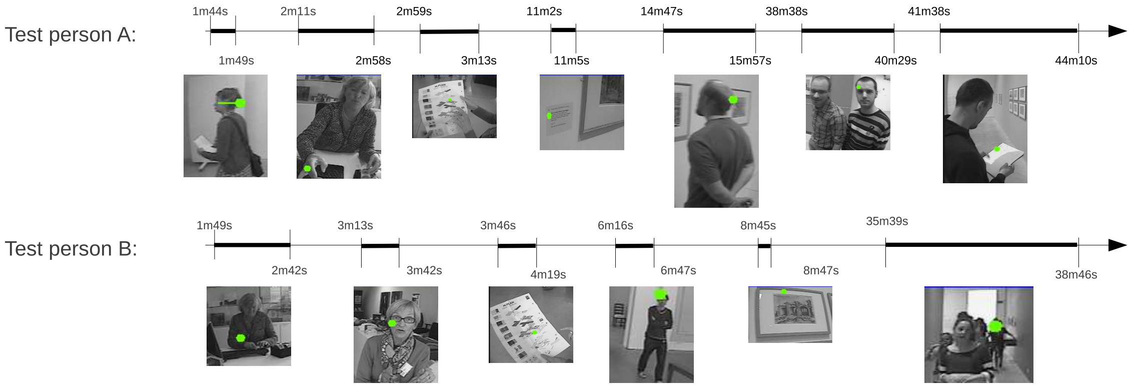

Figure 12: Results of our algorithm applied to the recordings of the museum visit. Each timeline represents a short summary of viewing behaviour of a participant.

REFERENCES

Alves, R., Lim, V., Niforatos, E., Chen, M., Karapanos, E., and Nunes, N. J. (2012). Augmenting customer journey maps with quantitative empirical data: a case on eeg and eye tracking.

Bay, H., Tuytelaars, T., and Van Gool, L. (2006). Surf: Speeded up robust features. In *ECCV*, pages 404–417.

Brône, G., Oben, B., and Feyaerts, K. (2010). Insight interaction. a multimodal and multifocal dialogue corpus.

Brône, G., Oben, B., and Goedemé, T. (2011). Towards a more effective method for analyzing mobile eye-tracking data: integrating gaze data with object recognition algorithms. In *Proc. of PETMEI*, pages 53–56.

Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). Brief: binary robust independent elementary features. In *ECCV*, pages 778–792.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893.

De Beugher, S., Brône, G., and Goedemé, T. (2012). Automatic analysis of eye-tracking data using object detection algorithms. PETMEI.

Dollár, P., Tu, Z., Perona, P., and Belongie, S. (2009). Integral channel features. In *BMVC*.

Dollár, P., Wojek, C., Schiele, B., and Perona, P. (2012). Pedestrian detection: An evaluation of the state of the art. *PAMI*, 34.

Evans, K. M., Jacobs, R. A., Tarduno, J. A., and Pelz, J. B. (2012). Collecting and analyzing eye-tracking data in outdoor environments. *Journal of Eye Movement Research*, pages 1–19.

Felzenszwalb, P. F., Girshick, R. B., and McAllester, D. (2010). Cascade object detection with deformable part models. In *CVPR*, pages 2241–2248. IEEE.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

Gall, J. and Lempitsky, V. (2009). Class-specific hough forests for object detection. In *CVPR*.

Henderson, J. M. (2003). Human gaze control during real-world scene perception. *Trends in Cognitive Sciences*, 7(11):498 – 504.

Jokinen, K., Nishida, M., and Yamamoto, S. (2009). Eye-gaze experiments for conversation monitoring. In *Proc. of the 3rd IUCS*, pages 303–308.

Judd, T., Ehinger, K., Durand, F., and Torralba, A. (2009). Learning to predict where humans look. In *ICCV*, pages 2106–2113. IEEE.

Leutenegger, S., Chli, M., and Siegwart, R. (2011). Brisk: Binary robust invariant scalable keypoints. In *ICCV*.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. In *IJCV*, pages 91–110.

Mathias, M., Benenson, R., Timofte, R., and Van Gool, L. (2013). Handling occlusions with franken-classifiers. Submitted for publication.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., and Van Gool, L. (2005). A comparison of affine region detectors. *IJCV*, 65(1-2):43–72.

Miksik, O. and Mikolajczyk, K. (2012). Evaluation of local detectors and descriptors for fast feature matching. In *ICPR*, pages 2681–2684.

Rosten, E. and Drummond, T. (2005). Fusing points and lines for high performance tracking. In *ICCV*, ICCV, pages 1508–1515.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *ICCV*, pages 2564–2571.

Toyama, T., Kieninger, T., Shafait, F., and Dengel, A. (2012). Gaze guided object recognition using a head-mounted eye tracker. In *ETRA*, pages 91–98.

Tuytelaars, T. and Mikolajczyk, K. (2008). Local invariant feature detectors: a survey. *Found. Trends. Comput. Graph. Vis.*, 3(3):177–280.

van Gompel, R. (2007). *Eye Movements: A Window on Mind and Brain*. Elsevier Science.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *CVPR*, pages 511–518.

Yun, K., Peng, Y., Samaras, D., Zelinsky, G. J., and Berg, T. L. (2013). Studying relationships between human gaze, description, and computer vision. In *CVPR*.