GazeVideoAnalyser: A Modular Software Approach Towards Automatic Annotation of Gaze Videos

Kai Essig, Dato Abashidze, Manjunath Prasad and Thomas Schack

1 Introduction/Related Work

With the development of mobile eye-tracking systems over the last years, eye movements can now be recorded when humans are in sensomotoric contact with their environment. The analysis of eye movements in natural scenes yields valuable insights into the cognitive processes underlying scene perception and to explore the strategies our visual system uses in the initiation and guidance of actions [Land and Tatler 2009; Evans et al. 2012]. The application of eye-tracking techniques in real-world conditions led to a bunch of new problems amongst others: participants perceive the world from different perspectives, the lighting conditions change, relevant objects move over time and may be partially or fully occluded. Furthermore, in order to track participants' gaze positions in dynamic environments, eye-tracking systems have to meet completely diverse demands, such as careful eye-tracker calibration, tracking of eye features and gaze data analysis [Evans et al. 2012]. Whereas the hardware of the mobile systems is quite developed, automated methods for calibration, pupil- and fixation detection, and gaze analysis in field studies need further research [Evans et al. 2012].

There exist already some approaches to overcome the time-consuming and error prone manual annotation process of gaze videos. [Land and Mc Leod 2000] determine the gaze angle by combining head- and eye-in-head orientation, calculated from the movement of fixated objects and the gaze cursor in the scene video. For dynamic applications the target must be manually coded in the video. Paletta et al. [2013] generate first a 3D model of the scene by using a Kinect and the RGB-DSLAM methodology. Their multi-component vision system then outputs the 3D point of regard, the gaze positions, frustrum and saliency map overlaid onto the acquired 3D model. This approach needs a tripod to hold the Kinect and HD camera and was applied to a supermarket scenario. Beugher et al. [2013] describe a system for the analysis of recorded gaze videos by combining trained object recognition, person- and facetracking algorithms. After the desired target is recognized, the fixated object is labeled. Two extensions to the system are suggested to improve the detection performance of faces and bodies: 1.) a human torso detector is trained on images from the VOC2009 dataset, 2.) the gaze cursor is used as a tracker to prevent false detections and to overcome missing detections.

Published at SAGA Workshop 2015, proceedings archived under DOI: 10.2390/biecoll-saga2015_0.
Individual article archived under DOI: 10.2390/biecoll-saga2015_2

The overview reveals that each solution has its own advantages and limitations. Either it 1.) is tailored to a particular setting (e.g. supermarket); 2.) is not applicable in unrestricted natural scenarios; 3.) needs data pre- or postprocessing; or 4.) only works with additional hardware. Furthermore, time-saving factors (e.g., ease of use, the ability to add AOIs (areas of interest) after the recording is finished, compatibility with statistic software) have to be considered [Evans et al. 2012].

2 Our Contribution

In order to provide a generally applicable and time-saving approach to the analysis process, we developed a modular software called GazeVideoAnalyser that allows for fully- and semiautomatic annotation of gaze videos recorded in unrestricted natural scenarios. The user can select target objects of interest by manually "roping" a rectangular lasso with the mouse around them at particular scene positions in the scene video provided by the mobile eye-tracking system. The selected part is then cut out of the video frame and responding feature vectors are calculated. The software overcomes the limits of existing, largely application specific solutions by combining different object recognition and tracking methods, without the need of any scene or data preparation (e.g., no markers or models are necessary). GazeVideoAnalyser provides a semi-automatic interpolation function, as well as Speeded-Up Robust Features (SURF) [Bay et al, 2006; Evens, 2009] and HSV Color Object Tracking [Smith and Chang, 1996]. Gaze information is used to tune tracking parameters (i.e., to weight fixated areas or to exclude false positives).

3 Discussion

In a preliminary evaluation study we compared the results of the two tracking algorithms against each other and to those of a manual annotation based on scene videos of a typical day-by-day task. Each recorded scene video (MPEG-4) has a resolution of 800 x 600 pixels at 25 fps and a duration of around 1 minute. The scene videos were manually labeled with the ELAN annotation software [ELAN Annotation Tool]. Additionally, all videos were analyzed fully-automatically using the GazeVideoAnalyser on a Quadro Core i7 -4810MQ CPU computer with 2.8 GHz and 8 GByte RAM. The manual annotation of each video took around 20-25 minutes, depending on its length. The automatic annotation lasts around 2-3 minutes (Color Tracker) and 6-7 minutes (SURF) (while feedback results were displayed online), resulting in significant time savings (factor up to 8).

Published at SAGA Workshop 2015, proceedings archived under DOI: 10.2390/biecoll-saga2015_0.
Individual article archived under DOI: 10.2390/biecoll-saga2015_2

All in all, the results indicate that the GazeVideoAnalyser provides a reliable automatic video analysis even under challenging recording conditions and can thus significantly speed up the annotation process. By providing no online feedback this factor can even be increased. Furthermore, the tracking algorithms have shown different performance advantages under diverse experimental conditions, making our modular approach with various tracking algorithms a suitable tool for the annotation of videos from natural scenes.

ACKNOWLEDGMENTS: This research/work was supported by the Cluster of Excellence Cognitive Interaction Technology 'CITEC' (EXC 277) at Bielefeld University, which is funded by the German Research Foundation (DFG).

References

1. Bay H, Tuytelaars T, Shah M (2006). Surf: Speeded up robust features. European Conference on Computer Vision, 1, 404-417.
2. De Beugher S, Youne I, Brône G, Goedemé T (2012). Automatic analysis of eye-tracking data using object detection algorithms. *In Proc. of ACM Conf. on Ubiquitous Computing*, 677-680.
3. Evans C. (2009). Notes on the OpenSURF Library. CSTR-09-001, University of Bristol, January 2009.
4. Evans KM, Jacobs RA, Tarduno JA, Pelz J. (2012). Collecting and Analyzing Eyetracking Data in Outdoor Environments. Journal of Eye Movement Research, 5(2):6, 1-19.
5. Land MF, Mc Leod P. (2000). From eye movements to actions: how batmen hit the ball. *Nature Neuroscience, 3*, 1340-1345.
6. Land MF, Tatler BW (2009). Looking and acting –vision and eye movements in natural behaviour. Oxford: University Press.
7. Paletta L, Santner K, Fritz G, Mayer H, Schrammel J (2013). 3D Attention: Measurement of Visual Saliency Using Eye Tracking Glasses, *Proc. CHI 2013.*
8. Smith JR, Chang, SF (1996). Tools and techniques for colour image retrieval. In I.K.Sethi & R.C. Jain (Eds.), Proceedings SPIE Storage and Retrieval for Still Image and Video Databases IV, 2670, 426–437.
9. ELAN – Annotation Tool: http://www.lat-mpi.eu/tool/elan/.

Published at SAGA Workshop 2015, proceedings archived under DOI: 10.2390/biecoll-saga2015_0.
Individual article archived under DOI: 10.2390/biecoll-saga2015_2