MutualEyeContact: A conversation analysis tool with focus on eye contact

Alexander Schäfer
TU Kaiserslautern
alexander.schaefer@dfki.de

Tomoko Isomura
Waseda University
Japan Society for the Promotion of Science
isomurat8818@gmail.com

Gerd Reis
German Research Center for Artificial Intelligence
gerd.reis@dfki.de

Katsumi Watanabe
Waseda University
University of New South Wales
katz@waseda.jp

Didier Stricker
German Research Center for Artificial Intelligence
TU Kaiserslautern
didier.stricker@dfki.de

ABSTRACT

Eye contact between individuals is particularly important for understanding human behaviour. To further investigate the importance of eye contact in social interactions, portable eye tracking technology seems to be a natural choice. However, the analysis of available data can become quite complex. Scientists need data that is calculated quickly and accurately. Additionally, the relevant data must be automatically separated to save time. In this work, we propose a tool called MutualEyeContact which excels in those tasks and can help scientists to understand the importance of (mutual) eye contact in social interactions. We combine state-of-the-art eye tracking with face recognition based on machine learning and provide a tool for analysis and visualization of social interaction sessions. This work is a joint collaboration of computer scientists and cognitive scientists. It combines the fields of social and behavioural science with computer vision and deep learning.

CCS CONCEPTS

• **Human-centered computing** → **Visualization toolkits**; *Scientific visualization*; • **Computing methodologies** → *Tracking*.

KEYWORDS

mutual eye contact, conversation analysis, eye tracking, visualization tool

ACM Reference Format:
Alexander Schäfer, Tomoko Isomura, Gerd Reis, Katsumi Watanabe, and Didier Stricker. 2020. MutualEyeContact: A conversation analysis tool with focus on eye contact. In *Symposium on Eye Tracking Research and Applications (ETRA '20 Short Papers), June 2–5, 2020, Stuttgart, Germany.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3379156.3391340

ETRA '20 Short Papers, June 2–5, 2020, Stuttgart, Germany
© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-7134-6/20/06...$15.00
https://doi.org/10.1145/3379156.3391340

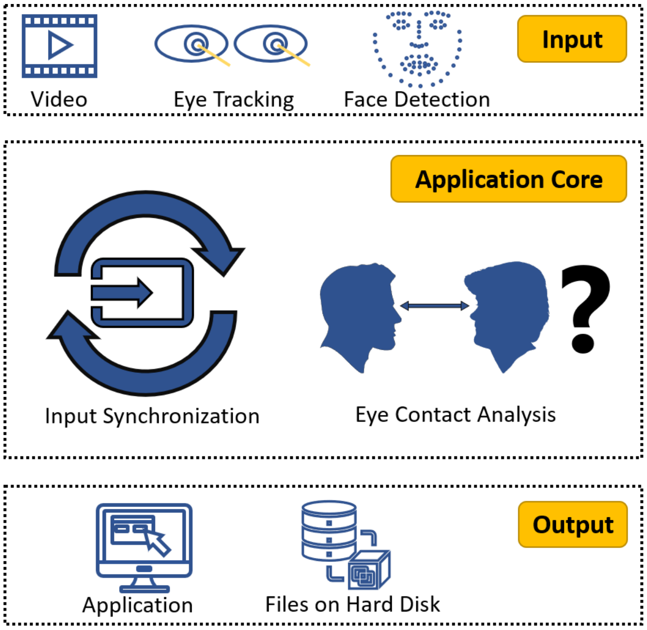

Figure 1: MutualEyeContact is a tool that combines state-of-the-art tracking and detection algorithms. Main focus is the analysis of conversations between two individuals in respect to eye contact.

1 INTRODUCTION

Scientists need to be able to analyse collected data efficiently and correctly in order to draw useful conclusions. In our case we are addressing the problem of efficiently analysing human behavior and emotions during social interactions between individuals. We want to analyse how the human body behaves during eye contact with another person and to the best of our knowledge, there is no specific tool available to analyse this case efficiently. MutualEyeContact was developed for this purpose (see Figure 1 for an overview). It supports scientists in understanding the behavior of the human body during natural social interactions. We combine and synchronize machine learning based face tracking with wearable eye tracking hardware.

Researchers have spent many hours analysing and labeling videos by hand because there is no tool available for their specific research problem. As an example, [Rogers et al. 2018] reported that manual coding of gaze behavior in 4-min conversations took them approximately 62 hours. It is worth mentioning that subjects need some time to get used to invasive tracking devices. This can lead to falsified data in the beginning of a recording session since participants are behaving unnatural due to the observation. To compensate for this falsified data, longer data recording sessions should be preferred. This increases the necessary manual labeling work even more. Therefore, short sessions for manual labeling tasks are currently preferred. By utilizing the technique of a vision-based face-tracking system, most of the time spent for this process can be omitted. Our tool is automated and therefore does not add any additional work depending on the amount of available data.

For eye tracking we are using Tobii Pro Glasses 2 (TPG2). OpenFace 2.0 toolkit from [Baltrusaitis et al. 2018] is used for facial landmarks and facial action unit recognition. Facial landmarks represent important regions of the face such as eyes, eyebrows, nose, mouth and jaw. The Facial Action Coding System (FACS) [Friesen and Ekman 1978] is a coded system to support emotion analysis. It assigns Action Units to almost every visible movement of the musculature used for face mimic.

Our work fosters in-depth analysis of data related to eye contact in natural social interactions. The contribution of this work can be summarized as follows:

- Significant time savings by using the presented tool for mutual eye contact analysis in respect to traditional methods
- More in-depth analytics for researchers due to automatic data extraction while analysing recorded footage with filter selection (e.g. mutual eye contact + Facial Action Units)

2 BACKGROUND AND RELATED WORK

Eye-gaze plays various roles in human social interactions. In particular, through the direct gaze which often results in eye-contact, we are able to perceive and signal a variety of meanings, such as intention to communicate or to exchange turns of speaker [Ho et al. 2015], threat and dominance [Emery 2000], interest [Argyle and Cook 1976], or seeking for approval [Efran 1968]. Recently, some theories propose that eye-contact activates social brain networks [Senju and Johnson 2009], and facilitates self-referential processing [Conty et al. 2016]. Furthermore, the latest hyper-scanning studies showed that eye-contact triggers neural coherence between agents [Hirsch et al. 2017], [Piazza et al. 2018], suggesting that eye-contact enhances the temporal alignment of two brains and facilitates the information sharing [Cañigueral and Hamilton 2019]. While socio-cognitive function of eye-contact has frequently been shown, many of these studies are conducted under experimentally manipulated settings, where for example the participants are required to intentionally fixate on the partner's eye region. One of the reasons for this is because tracking eye-gaze in natural human-to-human interactions is by no means easy. In real human-to-human interactions, such as dyadic conversation, eye movements of both agents are generally very active. Since both agents constantly alternate their gaze between eyes of their partner and other regions, the exchange of the eye-gaze happens very quickly. Accordingly, temporal and spatial resolution of measures is critical to address the interpersonal dynamics of gaze behavior in naturalistic settings.

Recently, several studies have tackled the nature of temporal dynamics of gaze in real human-to-human interactions by simultaneously measuring both agent's eye-gaze using wearable eye-trackers [Broz et al. 2012; Ho et al. 2015; Rogers et al. 2019, 2018]. In particular, Rogers et al. (2018) used eye tracking and attempted to code the temporal and spatial details of each agent's gaze patterns and its interpersonal interactions during a 4-min natural conversation, by analysing the time course of fixated facial area (e.g., forehead, eyes, nose, mouth). They showed that during the conversation the agents spent on average 60% of the time directing their gaze toward face of the other person, but only 10% of time directed specifically to the eyes; and that mutual face-gaze events were approximately 2.2 s long, but it was only 0.36 s long for the mutual eye-contact events. These results demonstrated that occurrence of mutual eye-contact is surprisingly momentary.

While the techniques of dual eye-tracking seemingly paves the way for cognitive scientists to further investigate the role of eye-contact in natural social interactions, load for manually coding the gaze allocation apparently prevents the efficient progress of the research. Wearable eye-tracking glasses usually have two cameras; one facing forward to record a video of wearer's field of vision; and one measures the wearer's eye gaze which is represented as a 2-D coordinate based on a frame-by-frame static image of the video. As the faces of both agents constantly move during a natural conversation, coordination of the face and features within the face do not remain in a fixed location in the video, meaning that gaze allocation should be identified frame-by-frame based on both video and gaze data. As dual eye-tracking requires twice of this work, a substantial amount of time is needed to manually code the gaze behavior of both agents.

3 IMPLEMENTATION

In this section we describe how MutualEyeContact is implemented. We explain how multiple systems are integrated and how the synchronization of video, eye tracker and face detection was done. The application allows to apply filters (e.g. mutual eye contact, eye contact) to automatically extract and display useful information from the input. Action Units (blinking, chin raiser, brow raiser etc.) provided by the face recognition algorithms can be extracted frame-by-frame as well (for a full list of available Action Units we refer the reader to [Baltrusaitis et al. 2018], [Baltrušaitis et al. 2015]).

Filters can be combined with each other to even further highlight certain aspects of input data. For example, it is possible to combine Action Unit filters and estimate emotions, e.g. *Cheeck Raiser + Lip Corner Puller* is considered as happiness emotion according to EMFACS (Emotional Facial Action Coding System [Friesen et al. 1983]). This filter could then be used find a correlation between eye contact and the happiness emotion.

3.1 Data Filtering

Most filters will process a whole video, extracting eye tracking and face recognition data at a given frame and calculate specific data with a specified output as shown in Figure 2. This will produce a frame-by-frame data output which can be used to analyse behaviour at certain areas of the input data. Some filters are used to combine others e.g. mutual eye contact filter uses two eye contact filters.

As an example, there is a recorded conversation between person A and person B. Both will have their eye gaze tracked by wearable eye tracking devices. Additionally to that, the outward facing camera of the eye tracker is recording the field of view of person A, which is used for face detection of person B and vice versa. If a scientist wants to know whether person A is looking into the eyes of person B, the eye contact filter can be applied to the input data. This filter will extract eye gaze position A_{eye} of person A and face position B_{face} of person B for each video frame by using the internal synchronization algorithms (see chapter 3.2). After A_{eye} is obtained, the pixel coordinate $P(x, y)$ of the eye gaze is obtained by re-projecting A_{eye} to the image plane of B_{face}. After that a point in polygon test with point $P(x, y)$ and the landmarks of B_{face} is done. This results in a *true* or *false* value for each frame, which is stored in an internal data structure. If required, a more descriptive and continous value in the range of 0 to 1 of eye/face contact can be calculated by using the known distance between gaze point and the region of interest (i.e. eye or face area).

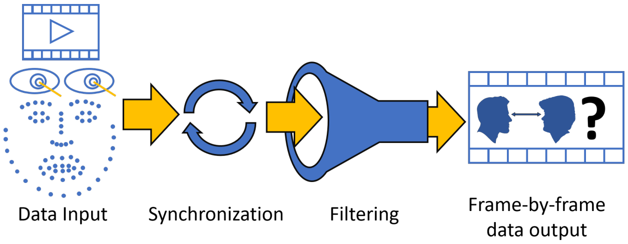

Figure 2: Schematic overview of our data processing pipeline. The tool takes video, eye tracking and face detection data as input. After the data is synchronized, filters can be applied to the data to highlight certain video parts.

After a filter is completed and has finished processing, it is displayed in a highly customizable timeline widget (see Figure 4).

3.2 Input Data Synchronization

As mentioned in chapter 2, face-gaze events are approximately 2.2s long, while mutual eye-contact lasts for 0.36s, represented by 9 consecutive frames in the video. Therefore, the synchronization deviation should not exceed 9 frames in case of a 25Hz video. With the approach described in this section, we can guarantee a synchronization accuracy of less than 3 frames. Eye trackers usually have a higher framerate than videos, in our case we used trackers with 50Hz. The eye tracker uses inward facing cameras for eye tracking and one additonal outward facing camera to show what the wearer is seeing. The eye tracking data is then re-projected to an image plane of the outward facing camera where the eye gaze is available as pixel coordinates. The data stream from the eye tracker needs to be synchronized to the outward pointing camera and any additional data (e.g. face recognition). The face recognition output from OpenFace toolkit uses OpenCV [Bradski 2000] in the backend and produces an output which labels each individual frame with a unique number. The eye tracking data stream consists of a timestamp T_e from the eye Tracker and a corresponding video time stamp T_{vts} which represents the current time on the tracker since the video started.

Since frame encoding/decoding does vary depending on the used codec and OpenCV not using video timestamps as frame descriptor, there is a descriptor mismatch problem which needs to be solved. Additionally, there is also a framerate mismatch, because the eye tracker is having a much higher framerate ($50Hz$) than the video ($25Hz$).

The eye tracker sends a periodic signal which contains a timestamp from the eye tracker T_e, a corresponding video time stamp T_{vts} and a presentation timestamp T_{pts}. Presentation timestamps (PTS) are used in video synchronization and denote a start time *PTSB* and an end time *PTSE* when an individual frame should be displayed. With this information, we can solve the mismatch problems with equation 1.

$$F_{number} = \frac{F_{ptse} - FF_{ptse}}{F_{dur}} \quad (1)$$

Here F_{ptse} denotes the PTS end time of the current frame, FF_{ptse} the PTS end time of the first valid frame in the video and F_{dur} denotes the time in microseconds a frame takes to be displayed.

We used the uncompressed video output from the eye tracker as input to our software to avoid unnecessary information loss due to encoding/decoding. For displaying the video frames we chose the Windows Media Foundation (WMF) library for video decoding since it provides accurate video timestamps for each frame.

3.3 Tool Overview

In this section we briefly introduce the graphical user interface (GUI) of the proposed tool as well as some suggestions which were implemented after testing the tool. The GUI consists of four main elements: data loading, video viewing, video controls and filters. The video viewing part of the video features a side-by-side view of two input videos as shown in Figure 3. Green dots in the area surrounding the face are facial landmarks detected by the face detector. A yellow circle represents the gaze point of the agent wearing the eye tracker.

After loading the necessary files and pre-processing (e.g. solving the synchronization problem), the controls for video manipulation are enabled. As described in the previous section, filters can be applied to the input data which is the core aspect of our tool. There can be multiple filters applied to the data, each represented by a new timeline widget as shown in Figure 4. Applying filters to the data consumes a lot of processing power, depending on how much data is available (e.g. a 20 minute video has approx. 30.000 frames at 25Hz). Processing time can vary from a few seconds to a few minutes depending on the length of a recorded session. For a video with about 30.000 frames, the eye contact filter requires about 0.5 seconds and the mutual eye contact filter about 1.5 seconds to compute (on an Intel Xeon E3-1245 v6 CPU). For this reason, each time a filter is applied, a new thread is created that processes the data simultaneously. While a filter is being processed, a content blocker is shown over the specific filter area. This enables a smooth non-blocking user experience for the user. Hardware acceleration for video display is used, since multiple videos in at least $1920x1080$ resolution can result in a staggering view experience.

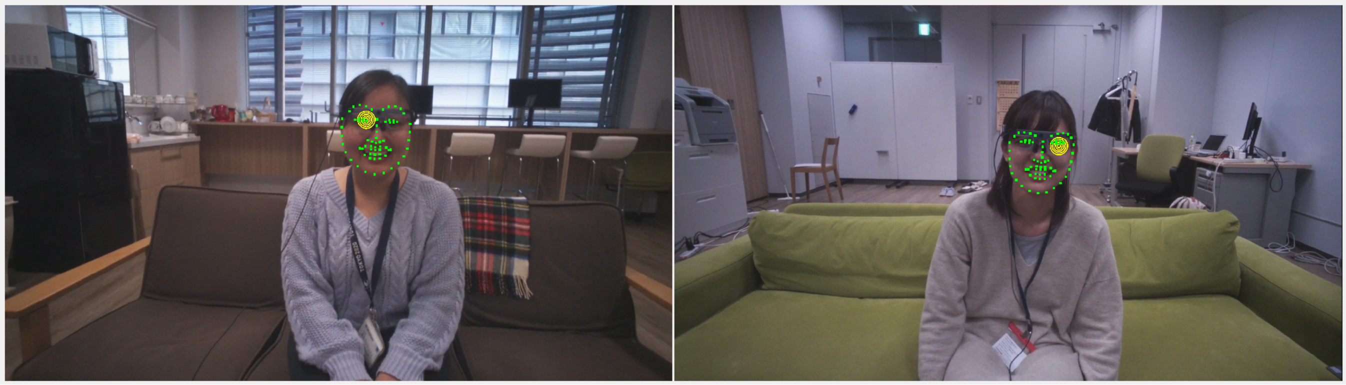

Figure 3: Side-by-side view of two input videos as shown in the proposed tool. The green dots are highlighting the facial landmarks detected by the face detector. The yellow circle represents the gaze point as tracked by the eye tracker

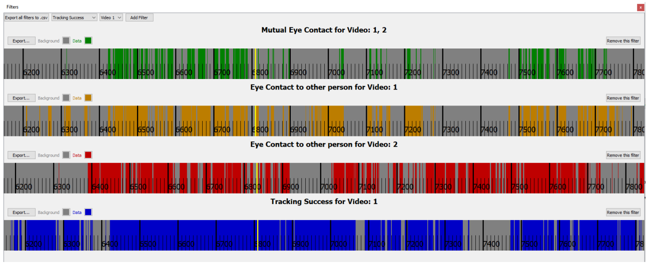
Filters
Export all filters to .csv
Tracking Success
Video 1
Add Filter
Mutual Eye Contact for Video: 1, 2
Export...
Background
Data
Remove this filter
6200
6300
6400
6500
6600
6700
6800
6900
7000
7100
7200
7300
7400
7500
7600
7700
Eye Contact to other person for Video: 1
Eye Contact to other person for Video: 2
7800
Tracking Success for Video: 1

Figure 4: Multiple filters applied to the input data, each represented in it's own timeline widget.

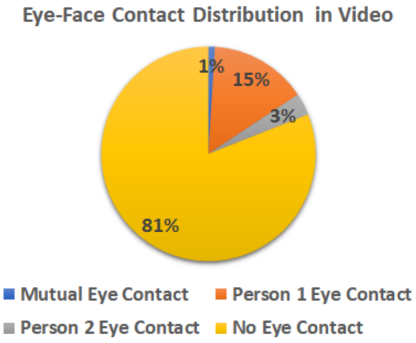

Figure 5: Eye-Face contact distribution during a recorded 20 minute social interaction session.

4 PILOT TESTING AND EXPERIMENTS

To evaluate our tool, computer scientists as well as cognitive scientists tested the application. We previously recorded multiple social interaction sessions as data input. To record this data, participants were told to sit in front of each other with a distance of about 3 meters. Each participant was wearing an eye tracker to record the eye gaze. The outward facing camera of the eye tracker recorded the field of view of the participant. The participants were told to have a conversation for about 20 minutes. After that the session closed and the data was stored.

We used our tool to explore certain data during the social interaction sessions. As an example, we calculated the eye-face contact distribution which is shown in Figure 5. This data is a combination of several filters i.e. *mutual eye contact*, *eye contact person 1* and *eye contact person 2*. It should be mentioned that face recognition does not work in all circumstances (e.g. due to occlusion) and frames without valid face recognition are therefore omitted. While a manual dissection of this input data would take many hours, our tool requires only 1-3 seconds to compute.

5 CONCLUSION AND FUTURE WORK

In this paper, we proposed a tool for researchers to analyse the various roles of eye contact in natural social interactions. By combining reliable eye tracking with face detection we are able to create a useful tool for scientists that can save many hours of labeling and analysing videos manually. Wearable eye tracking devices are used for reliable and accurate eye-gaze tracking. We use facial landmark and facial Action Unit recognition together with eye-gaze tracking and combine it into one system. This tool fills the gap between the fields of social and behavioural science, computer vision and machine learning to enable a powerful in-depth analysis for eye contact related research problems. In the future, this work can be extended with features like remote photoplethysmography for non-invasive vital sign monitoring during social interactions (e.g. heart rate, respiration rate). We also plan to include monocular full body 3D pose estimation as described in the work of [Kovalenko et al. 2019] to even further support the investigation of the human body during social interactions. In future work, we intend to use this tool to conduct (mutual) eye contact research studies.

ACKNOWLEDGMENTS

This work was partially funded by: Offene Digitalisierungsallianz Pfalz which is part of the Innovative Hochschule (Grant No: 03IHS075B); CREST, Japan Science and Technology Agency (Grant No: PMJCR14E4); KAKENHI, Japan Society for the Promotion of Science (Grant No: 17H06344 and 19K20651); VIDETE, German Federal Ministry of Education and Research (BMBF) (Grant No. 01|W18002).

REFERENCES

Michael Argyle and Mark Cook. 1976. Gaze and mutual gaze. (1976).

Tadas Baltrušaitis, Marwa Mahmoud, and Peter Robinson. 2015. Cross-dataset learning and person-specific normalisation for automatic action unit detection. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, Vol. 6. IEEE, 1–6.

Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. 2018. Openface 2.0: Facial behavior analysis toolkit. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, 59–66.

G. Bradski. 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000).

Frank Broz, Hagen Lehmann, Chrystopher L Nehaniv, and Kerstin Dautenhahn. 2012. Mutual gaze, personality, and familiarity: Dual eye-tracking during conversation. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 858–864.

Roser Cañigueral and Antonia F de C Hamilton. 2019. The role of eye gaze during natural social interactions in typical and autistic people. *Frontiers in Psychology* 10 (2019).

Laurence Conty, Nathalie George, and Jari K Hietanen. 2016. Watching eyes effects: When others meet the self. *Consciousness and cognition* 45 (2016), 184–197.

Jay S Efran. 1968. Looking for approval: Effects on visual behavior of approbation from persons differing in importance. *Journal of Personality and Social Psychology* 10, 1 (1968), 21.

Nathan J Emery. 2000. The eyes have it: the neuroethology, function and evolution of social gaze. *Neuroscience & biobehavioral reviews* 24, 6 (2000), 581–604.

E Friesen and Paul Ekman. 1978. Facial action coding system: a technique for the measurement of facial movement. *Palo Alto* 3 (1978).

Wallace V Friesen, Paul Ekman, et al. 1983. EMFACS-7: Emotional facial action coding system. *Unpublished manuscript, University of California at San Francisco* 2, 36 (1983), 1.

Joy Hirsch, Xian Zhang, J Adam Noah, and Yumie Ono. 2017. Frontal temporal and parietal systems synchronize within and across brains during live eye-to-eye contact. *Neuroimage* 157 (2017), 314–330.

Simon Ho, Tom Foulsham, and Alan Kingstone. 2015. Speaking and listening with the eyes: gaze signaling during dyadic interactions. *PloS one* 10, 8 (2015), e0136905.

Onorina Kovalenko, Vladislav Golyanik, Jameel Malik, Ahmed Elhayek, and Didier Stricker. 2019. Structure from Articulated Motion: Accurate and Stable Monocular 3D Reconstruction without Training Data. *Sensors* 19, 20 (2019), 4603.

Elise A Piazza, Liat Hasenfratz, Uri Hasson, and Casey Lew-Williams. 2018. Infant and adult brains are coupled to the dynamics of natural communication. *bioRxiv* (2018), 359810.

Shane L Rogers, Oliver Guidetti, Craig P Speelman, Melissa Longmuir, and Ruben Phillips. 2019. Contact is in the eye of the beholder: the eye contact illusion. *Perception* 48, 3 (2019), 248–252.

Shane L Rogers, Craig P Speelman, Oliver Guidetti, and Melissa Longmuir. 2018. Using dual eye tracking to uncover personal gaze patterns during social interaction. *Scientific reports* 8, 1 (2018), 4271.

Atsushi Senju and Mark H Johnson. 2009. The eye contact effect: mechanisms and development. *Trends in cognitive sciences* 13, 3 (2009), 127–134.