This is a preliminary version of an article published by
Kopf, S., Schön, D., Guthier, B., Rietsche, R., Effelsberg, W.:
A Real-time Feedback System for Presentation Skills.
Proc. of World Conference on Educational Multimedia, Hypermedia and Telecommunications (EdMedia), pp. 1633-1640, June 2015.

A Real-time Feedback System for Presentation Skills

Stephan Kopf, Daniel Schön, Benjamin Guthier, Roman Rietsche, Wolfgang Effelsberg
Department of Computer Science IV
University of Mannheim
Mannheim, Germany
{kopf, schoen, guthier, rietsche, effelsberg}@informatik.uni-mannheim.de

Abstract: Giving a presentation is an everyday skill in many people's educational and professional life like school homework, college presentations, customer promotions, or financial reports. However, training is still rare and expensive. Important aspects like talking speed or body language are well understood, and many good practices exist, but they are difficult to evaluate. They require at least one experienced trainer who attends the presentation and is able to evaluate and give a constructive feedback. Our aim is to make a first step towards an automatic feedback system for presentation skills by using common motion-detection technology. We implemented a software tool using Microsoft's Kinect and captured gestures, eye-contact, movement, speech, and the speed of slide changes. A short evaluation using eight presentations in a university context showed that speaker movement and body gestures are detected well while not all spoken words and slide changes could be recognized due to the Kinect's limited technical capabilities.

Introduction

Giving a presentation is an important skill in many areas. However, it does not come naturally to all people and often needs a lot of training. Such presentation training is time-consuming and expensive. The speaker needs to pay attention to many different aspects, e.g.:

- Does the presenter's hand position look natural?
- How fast should the presenter move around?
- Is the talking speed appropriate?
- In which direction does the presenter look?
- For how long should a presentation slide be shown?

Depending on the density of information on the slides, the audience needs sufficient time to consume and understand it. Other obvious errors include a speaker talking too fast or jumping forwards and backwards between the slides during a presentation.

A general problem is the fact that presentation skills are taken for granted and usually not taught at the university (Pabst-Weinschenk, 1995), unless soft skill seminars are directly included in the degree program. Habits like making specific gestures or the way of speaking are permanent features and cannot be changed easily. Being nervous while speaking in front of a full audience is a common human behavior and makes presentation training even more difficult. Furthermore, students with less experience in presentation techniques often dare not to speak and practice in front of the class.

Audience response systems (e.g., quiz tools) are widely used in higher education today (Schön, 2012a). Students answer specific questions and the system provides direct feedback about the learning success. Although the flexibility of audience response systems increased a lot during the last years (Schön, 2012b), it is not possible to use these systems to evaluate a student's presentation.

Another approach could be to use a camcorder to record the presentation and provide automatic or manual feedback afterwards. Video annotation tools may be used offline by an experienced trainer to provide manual annotations of a talk (Kopf, 2012; Wilk, 2013). This approach provides high quality feedback, but the costs would also be very high. Automatic lecture recording systems are widely used in classrooms today and support all steps from capturing different video streams to publishing the content (Lampi, 2008a; 2008b). Although these systems automatically track the speaker, they do not provide feedback about the quality of a presentation. In previous work, we have developed algorithms for detecting and analyzing important regions (e.g., objects or people) in videos (Kopf, 2005; Richter, 2001) and used this information to resize a video (Kopf, 2011) or to select the most important temporal video segments (Kopf, 2004). These algorithms are not applicable because the computation time is too high and real-time feedback is not possible.

The idea behind our approach is to provide students of all levels and young professionals our real-time feedback system as a low-cost presentation trainer. The practice session can be carried out in a safe environment in which shy people feel more comfortable. The presenter gets direct feedback on their presentation style, including movement, gestures, and the duration of looking away from the audience. In addition, the system gives feedback on the presenter's rate of speech and the duration that each slide is shown. The presenter may also study the recorded audio-video presentation in detail and identify ways to improve speech, gestures and slides.

The paper is structured as follows: The next section gives a short overview of the underlying work in the areas of presentation skills and systems that analyze gestures by using the *Microsoft Kinect*. The following two sections present the details of our feedback system and evaluate it. The paper finishes with a conclusion and an outlook.

Related Work

Our feedback system for presentation skills focuses on five aspects: gestures, eye contact, pose and movement, speech, and slide changes. We first specify the requirements of good talks and derive objective parameters for each aspect. Our system computes these parameters by analyzing and aggregating the data from the input sensors and displaying it to the user. We also discuss related work that uses a Kinect as an input sensor to operate different multimedia systems.

Requirements of good talks

Hand and arm gestures during a presentation should appear natural and fit into the context. Speakers with less training often are unsure where to put their hands, whom to look at and where to stand. There are also some gestures that do not contribute to the presentation and should be avoided in most situations. Hand gestures below hip height are too small and restricted while gestures above the throat hinder eye contact (Hey, 2011). Furthermore, gestures such as crossing arms and hands in the pockets, or arms behind the back lead to a closed posture and block natural movements (Pabst-Weinschenk, 1995; Mentzel, 2008). Gestures with hands between hip height and chest height are indicated as positive (Mentzel, 2008).

Eye contact is very important in presentations and draws attention to the speaker. Especially in western cultures, eye contact is the main element of body language (Hey, 2011). The eyes of the dialog partner show if they are open, attentive, interested, skeptical, bored, or annoyed. The speaker can get information from the audience concerning possible questions, attention, and curiosity for more details. A good speaker absorbs this information and reacts accordingly (Hey, 2011). However, for speakers with less practice, observing the audience is perceived as too much information. Therefore, students typically look at the wall, at a display, or out of the window (Hey, 2011). This may not always be seen as a negative by the audience. According to Hey (Hey, 2011), the time a speaker looks into the same viewing direction should be between two to five seconds. Nöllke (Nöllke, 2011) states that longer periods (between 10 and 30 seconds) of looking away are acceptable when working with flip charts or pin boards.

Other important characteristics of a good presentation are *pose and movement* of the speaker. Hey, Nöllke and Pabst-Weinschenk (Hey, 2011; Nöllke, 2011; Pabst-Weinschenk, 1995) point out, that it is important to have a

good foothold. The speaker should move from time to time to make the presentation more vivid. But Pabst also observed that too much movement leads to disturbances. One of the reasons why inexperienced speakers often do not change their position are the closed gestures mentioned above.

Besides body language, *oral communication* is the most important factor in a presentation. The audio is significant for conveying information to the audience. Reasons for speaking too fast are a short amount of time, nervousness, and uncertainty about the subject (Mentzel, 2008). The speaker tries to speak fast to convey as much information as possible, but the absorbing capacity of the audience is limited. After exceeding the capacity, the listener usually stops following and goes absent-minded (Mentzel, 2008). Hence, it is critical for the speaker to take breaks in the flow of speech. This gives the audience the possibility to process what has been said so far and to re-think if the content was understood. Breaks are also beneficial to the presenter as the time can be used to think about the next sentence, the bridge to the next subject or to re-concentrate (Mentzel, 2008). Moreover, breaks are rhetorical devices which can be used to create tension, prepare a main hypothesis or to create silence at the beginning of the speech. According to Pabst-Weinschenk (Pabst-Weinschenk, 1995), the normal rate of speech is up to 140 words per minute. Mentzel (Mentzel, 2008) suggests a speech rate between 100 and 130 words per minute.

Time management is another issue many students have problems with. Nearly everyone has attended a speech, where a speaker significantly exceeded the set time limit. This usually happens when several talks happen at the same time, for example at conferences (Hey, 2011). The typical duration of a student's presentation is between 15 and 20 minutes. That puts them under an enormous pressure to fill the time with as much content as possible (Hey, 2011). A break between two words is seen as a waste, even though these breaks are an important factor of a good presentation. The audience needs time to process all the information and to understand what the speaker is talking about. This should lead to an optimal amount of slides for a time restricted presentation. A rule of thumb states that the speaker should at least talk for one minute per slide (Hey, 2011; Nöllke, 2011).

Kinect as input sensor

We use the *Microsoft Kinect* as input sensor. It includes an RGB camera, an infrared sensor and an infrared light source. The Kinect SDK provides a depth map of the scene and robust algorithms for estimating the 3D coordinates of body joints (Shotton, 2011). The Kinect also provides speech recognition based on control commands or free-text. Free-text speech recognition has the drawback that the software needs to be trained in order to recognize a particular voice and to improve its accuracy.

Using a Kinect for gesture recognition has attracted much research in various fields (Han, 2013; Wang, 2015; Yao, 2014; Dondi, 2014; Kim, 2015; Ren, 2013; Yang, 2013; Shum, 2013). Rahman et al. have developed a system that allows a simple and intuitive interaction with the in-car multimedia devices (Rahman, 2011). By using hand gestures, the interaction with the car while driving is reduced as much as possible. The captured gestures are then used in order to control the in-car multimedia devices, with which for example the media playlists can be browsed or the track of the audio player can be changed. For safety reasons the whole systems forgoes any graphical user interface which has to be operated while driving. In order to provide feedback to the driver, the system uses haptic signals and audio output.

Panger created a system which takes the Kinect into real-life kitchens (Panger, 2012). In a kitchen, a natural user interface can be very handy when the hands are dirty, covered by oven gloves or when hands are full. The presented system contains a recipe navigator, a timer, and a music player. All functions are controlled by gestures. Gallo et al. proposed another system which uses a Kinect as a human computer interface (Gallo, 2011). They propose an open-source system for controller-free, highly interactive exploration of medical images in critical medical environment such as an operating room. One of the main challenges in designing an interface for those fields is that the surgeons need to browse through the scans without having to physically touch any controller, since they need to stay sterile. Moreover, the interface needs to be easy to use in order to avoid time-consuming training to learn the interface.

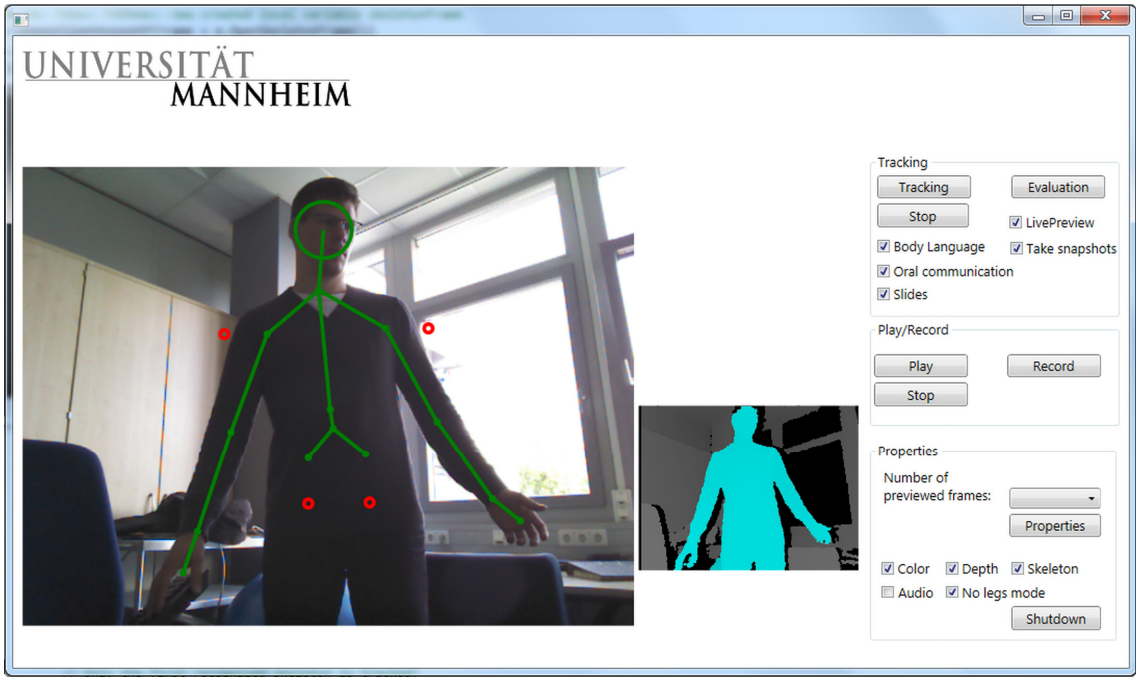

Figure 1: GUI (main view) of the feedback system. This view is visible when starting the feedback system. It allows to start, stop, or playback a recording as well as evaluate it.

In the above multimedia applications, the Kinect sensor is used to provide a touchless user interface. In contrast, our system does not support user interaction but allows the analysis, recognition, and visualization of complex behavior patterns of people's presentations.

Functionality of the Feedback System

Our system uses a Kinect as input sensor and supports functionality like capturing, storing and loading of the raw data streams, automatic analysis of human behavior, and visualization of the analyzed results. Figure 1 shows the graphical user interface of the system.

Raw data processing

The Kinect SDK offers high level functionality for human pose estimation and speech recognition[1]. To estimate a pose, depth information is computed and individual body parts are identified (Shotton, 2011). The Kinect SDK identifies up to 20 body parts for each person. The 3D positions of the joints of body parts define specific gestures.

The Kinect has four microphones and provides two speech recognition engines. The first one recognizes commands in order to control the Xbox, and the second one recognizes free-text such as dictation. One of the drawbacks of free dictation is that the software needs to be trained in order to recognize a particular voice and improve its accuracy. We require free-text recognition in order to investigate the speaker's voice but want to avoid a training phase. Therefore, we choose the Microsoft speech recognition engine *System.Speech*[2] instead

[1] Microsoft: Kinect for Windows. URL: https://msdn.microsoft.com/en-us/library/jj131033
[2] Microsoft: System.Speech. URL: https://msdn.microsoft.com/en-us/library/hh361625

of the Kinect speech recognition. Although the accuracy when combining the Kinect with Microsoft *System.Speech* is not very high, it is still sufficient for counting words.

Capturing the raw data streams is started manually by pressing the *tracking* button. RGB color values are stored with 8 bits per pixel whereas 16 bit values are used for depth information. Both streams are stored with a resolution of 640 x 480 pixels and a frame rate of 30 frames per second. Additionally, the audio stream captured from a four microphone array (16 kHz, 24 bit per sample) and skeleton information of each frame are stored in separate streams. For each stream element, additional attributes like frame number, time-stamp and type of stream are stored. The streams are saved to the hard-drive and can be replayed at any time.

The *replay method* is more complex due to possible synchronization errors. These errors may occur if frames are dropped during the capturing process or if the analysis of poses or the speech does not allow processing in real-time. While the color and skeleton frames only take 3 milliseconds to decode and display, decoding a depth frame takes 10 milliseconds. The reason for this behavior is that three of the bits of each pixel in the depth frame are reserved for encoding silhouettes of different persons which are filtered and colored before visualization. For playback, the different streams are decoded in parallel threads. They are synchronized by comparing the time-stamps with a centralized timer. If threads cannot be processed in real-time or if they sleep too long, the human eye immediately recognizes that the movement of the body is too slow. Furthermore, the overall time for the stream changes and consequently the tracking accuracy drops.

Estimation of the speaker's behavior

Robust gesture recognition is the fundamental module of our feedback system. Three essential behaviors of the speaker are identified:

- open gestures,
- hands below the waistline, and
- viewing direction of the speaker.

The position information provided by Kinect SDK is not always precise. Especially the shoulder and hip position may differ from physical coordinates. As can be seen in Figure 1, the green skeleton lines at the center of the person do not represent the correct hip position. A lower accuracy typically occurs when parts of the body like arms or legs are occluded or outside the viewing area. To overcome the problem of incorrectly detected positions of the shoulders or the hip, we move each shoulder coordinate horizontally until its position is no longer located within the shape of the person. Two thresholds are computed by slightly shifting both shoulder positions to the outside. Open gestures are identified when both hands are located outside of these threshold positions. Considering the hip, the larger the amount of cropping of a leg is the more the hip position is shifted down. The small red circles in Figure 1 specify the new positions of the shoulders and the hip.

The distance of the right and the left shoulder to the Kinect sensor is used in order to find out the *viewing direction of the presenter*. We make the assumption that the distance of both shoulders to the camera is very similar if the speaker is looking towards the camera. In the case that the distances of both shoulders to the camera differ, it is assumed that the presenter does not look at the audience anymore. This simple approach works well in most cases but errors occur if the speaker only turns the head to the wall and does not turn the shoulders. In our experiments, an automatic face detection algorithm was able to reduce these errors, but the increased computational cost prohibited real-time processing. We therefore decided to accept errors caused by head rotation.

Open gestures and *hands below the waistline* are detected by comparing the tracked positions of the hands to the shoulder and hip positions. When the speaker directly looks at the camera, a fixed horizontal offset is used to shift the shoulder position towards the outside. This offset reduces the number of incorrect detections of open gestures which are caused by measurement inaccuracies of the sensor. In case of a rotation of the speaker, both positions are shifted to the left or right, respectively.

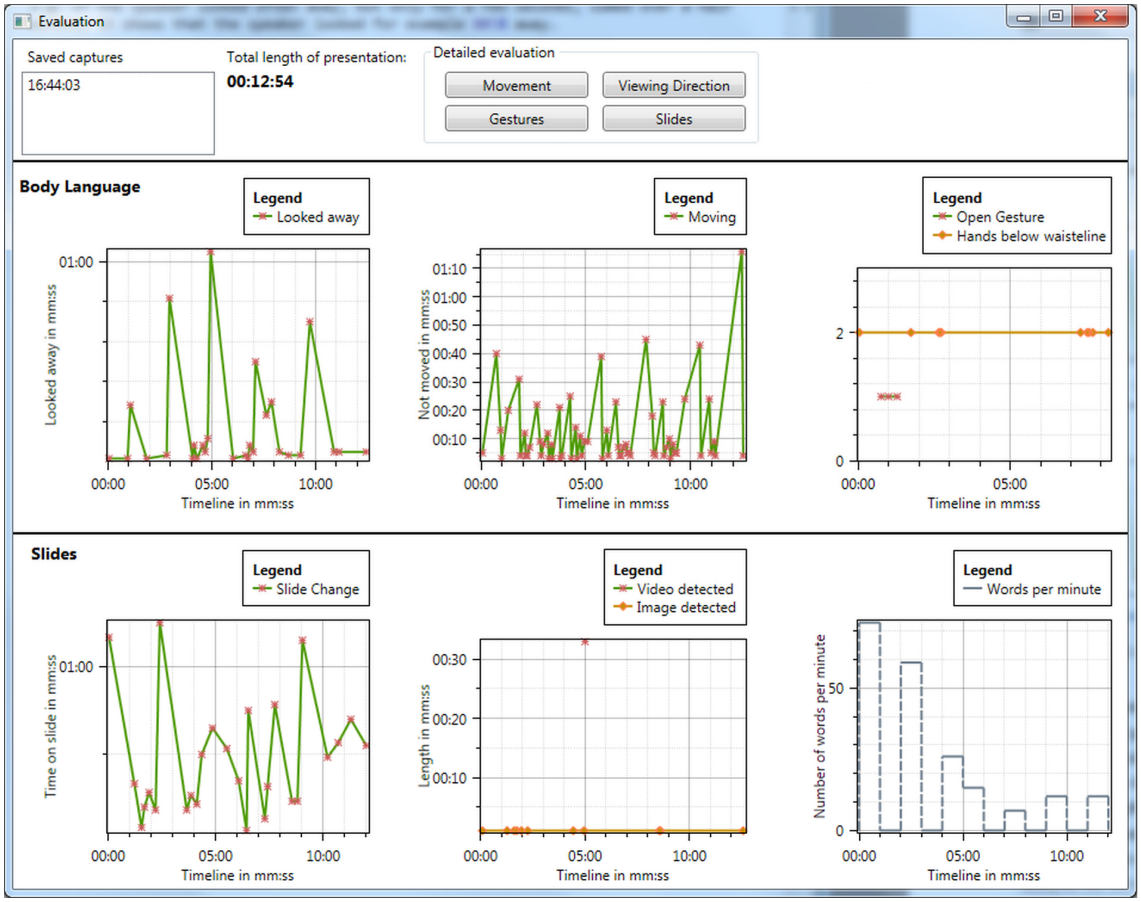

Figure 2: Overview of the feedback provided by the system. The graphs show a simulated talk that was captured to measure the accuracy of the system. Notice that the speaker was silent at predefined intervals, spoke a predefined number of words in other intervals, looked away at certain points in time, or started a video at minute 5:00.

Analysis of slides and speech

As visualized in Figure 3, the quality and pixel resolution of the RGB sensor are too low to reliably identify slide numbers. Instead, we measure the amount of pixel changes by computing the sum of absolute pixel differences. Histogram differences cannot be used due to the fact that background colors and text colors typically do not change between two slides. Even using absolute pixel differences, the problem remains that the speaker moves and may occlude the screen. This causes significant pixel changes. This error is avoided by computing the silhouette of the speaker and ignoring occluded pixels (see cyan colored area in Figure 1).

The Microsoft speech engine is used to recognize words. Because it is not our goal to analyze the content of a presentation, the number of recognized words within a given time interval is the only relevant information we compute. This information is processed in real-time and stored in a database for visualization at a later time.

Graphical User Interface

The main view of the graphical user interface (see Figure 1) visualizes the captured data on the fly and allows for easy configuration of the parameters of the used algorithms. Functionality for *recording* and *playing* streams is available as well as *tracking* and *preview* options. When pressing the *evaluation* button, the current

presentation is analyzed and feedback about the slide changes as well as gestures, movements, speech, and viewing direction of the presenter are shown over time.

Figure 2 gives an overview of the main evaluation categories and the temporal graphs that may indicate possible problems of the current talk. As an example, the speaker looked away from the audience for more than one minute at time 05:00 minutes. Details may be visualized for each category. Figure 3 shows detailed information about slide changes. It can be seen that the speaker changed the slides very quickly between 1:30 and 2:00 minutes.

The user can browse each time line or zoom-in to look at details. By clicking on a graph, the video jumps to the corresponding frame and shows the behavior of the user at this point in time.

Evaluation

We carried out an evaluation with students in order to measure the accuracy of the presentation feedback tool. Therefore, we captured five seminar talks, as well as one Bachelor and two Master final thesis presentations. During the eight presentations, the 20 members of the audience (five Bachelor students, seven Master students, five PhD students, two Post-Docs, and a full professor) filled out feedback sheets and evaluated each speaker. The feedback sheets contained several categories of questions and a time line from 0 to 25 minutes for each category.

Presentation	1	2	3	4	5	6	7	8
Open gestures [per minute]	6.3	3.6	6.4	15.6	13.8	8.3	14.5	20.2
Hands below waistline [per minute]	1.3	6.1	4.0	4.3	1.5	5.8	1.8	2.1
Time looking away [duration in %]	22	3	21	11	4	15	5	8
Mean duration of slide [seconds]	35	26	54	49	93	81	42	44
Duration of presentation [minutes]	22	29	39	32	31	19	27	30
Spoken words [per minute]	159	188	162	118	157	138	152	166

Table 1: Automatic analysis of the evaluated presentations.

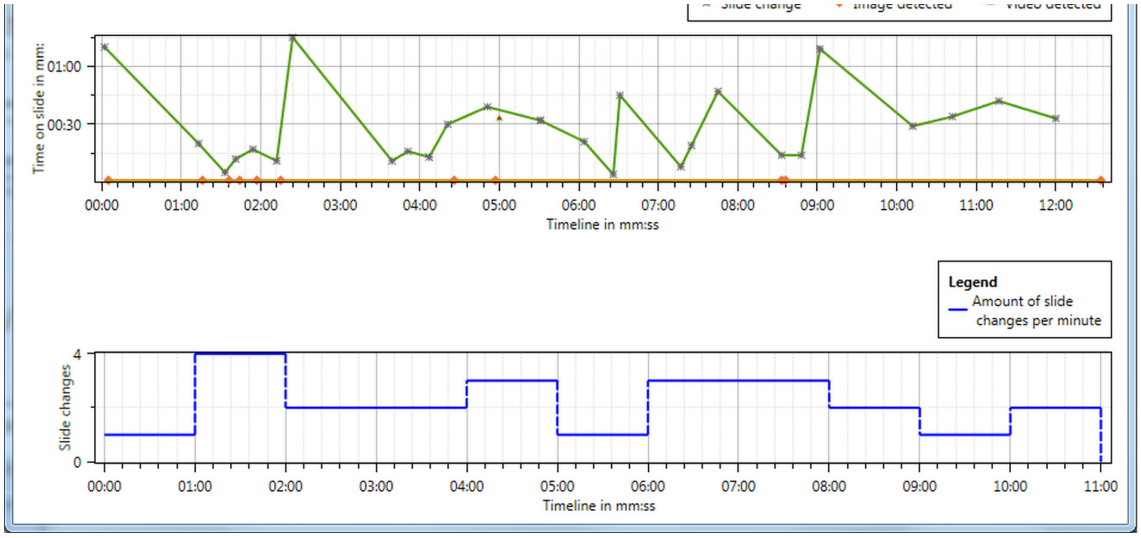

Figure 3: Detailed view about slide changes. The upper graph (green line) visualizes the time of a slide change and the duration each slide is presented. The lower graph (blue line) shows the number of slide changes within each interval (one minute).

The first column of Table 1 shows these categories. For each minute and category, the participants could mark negative observations, e.g., when a speaker looked away from the audience for a longer period of time. The numbers in Table 1 show the automatically estimated values computed by our evaluation tool. In the following, we will discuss the accuracy of the evaluation tool and how well the automatically estimated values correspond with the manual feedback from the sheets.

The number of automatically captured *open gestures* shows that presenters four, five, seven, and eight gesture a lot. When investigating the feedback sheets, several participants indicated that presenter four and five used too many gestures during the presentation. Presentation eight has been indicated by the system as the presentation with the largest number of open gestures. This correlates to the comments of the feedback sheets, e.g., *student appears very hectic*, *shows too many gestures*, or *gestures are often not appropriate*. For presentation seven, it was stated that there were good hand gestures and that they were not in excess. The speakers of the presentations one, two, three, and six do not use so many gestures. But a low number of gestures was not seen as a bad behavior in the evaluation sheets.

Presentation two stands out when considering the category *hands below the waistline* (more than 6 times per minute). This corresponds to the observations of the participants which clearly confirm this behavior. E.g., this negative observation was indicated 26 times on the time line and mentioned several times in the comments. The number of hands below the waistline are negatively correlated to the number of open gestures ($\rho = -0.49$).

Another aspect concerning the body language is the duration a speaker *does not look towards the audience*. The system indicates that the speakers in presentation one, three, and six have the highest overall time of looking away (between 15% and 22% of the presentation time). The feedback sheets agree with these results. E.g., the participants observed 55 times that speaker three is looking away from the audience.

The *mean duration each slide is shown* is one of the more challenging factors to compute automatically. Often, slide changes were triggered by animations or when a video was shown. Thus, the automatically detected slide changes have a high amount of false positives. Nevertheless, the data still provides some hints concerning if a speaker spends too much or insufficient time on a slide. Considering presentations one and two, the mean time a slide is shown is 35 and 26 seconds, respectively. This is significantly lower compared to the recommended time of at least 60 seconds per slide (Hey, 2011; Nöllke, 2011). According to the participants who filled out the feedback sheets, speaker eight did not spend sufficient time on each slide. The result was not so distinct for speakers one and two. In contrast, the speaker of presentation five clearly spends more time on each slide (93 seconds on average). By analyzing the sheets, we conclude that specifying an optimal duration for each slide is highly subjective and also depends on the content of each slide.

The *overall time for each presentation* was constrained to be between 25 and 30 minutes. Presentations one and six were obviously too short and presentation three was too long. This was also marked in the sheets. E.g., 57% of the participants stated that presentation three was too long. Additional comments were given like *too much information on the slides* or *the speaker had bad time management*. Thus, both the automatic feedback and the feedback sheets are consistent.

The last category considers the number of *spoken words per minute*. Again, the automatically computed data has a high error rate due to the untrained speech recognition. In our scenario, only the detection of spoken word boundaries, but not the correct recognition of words was relevant. The results are sufficiently precise to indicate whether a presenter speaks slowly or fast. Considering the recommended speaking rate of 130 words per minute, all speakers except speaker four exceed this value. The feedback sheets indicate that the rate of speech of speaker five was much too fast (measured 157 words per minute). Comments about the speaker were given like *speaking is very hectic* or *voice is fickle*. In contrast, speaker four was rated as having a clear speech and good pronunciation (118 words per minute). Speaker two had the highest rate of speech (188 words per minute) and spent less than half a minute on each slide. The participants observed that speaker two *hurried through the slides* and *did not plan the time flow very well*. These observations are clearly visible from the automatic feedback system as well.

Conclusion and Outlook

We have implemented a system for recognizing behavior patterns of presenters. We demonstrated that the system able to recognize basic presentation behaviors like hand gestures, eye contact, oral communication, and the speed of the visual presentation. Our system runs in real-time and allows recording and playback of RGB, depth, skeleton, and audio streams. A graphical time line shows the computed values of each category and allows the identification of possible improvements of a presentation.

In the near future, we would like to extend the system for providing feedback about the quality of the slides. An example could be to analyze if too much text is visible on a slide or if multimedia content (images, videos, animations) should be added to a presentation. More work needs to be done in order to automatically evaluate a presentation entirely. Aspects like slide design or nervousness of the speaker are difficult to track.

References

Dondi, P., Lombardi, L., & Porta, M. (2014). Development of gesture-based human–computer interaction applications by fusion of depth and colour video streams, *IET Computer Vision* (Volume 8, Issue 6, 568-578)

Gallo, L., Placitelli, A., & Ciampi, M. (2011). Controller-free exploration of medical image data: Experiencing the Kinect, in: *Computer-Based Medical Systems*.

Han, J., Shao, L. Xu, D., & Shotton J. (2013). Enhanced computer vision with Microsoft Kinect sensor: A review, *IEEE Transactions on Cybernetics* (Volume 43, Issue 5).

Hey, B. (2011). *Presenting in science and research* (in German), Berlin, Heidelberg: Springer.

Kim, Y., Lee, M., Park, J., Jung, S., Kim, K., & Cha, J. (2015). Design of Exhibition contents using swipe gesture recognition communication based on Kinect, *International Conf. on Information Networking* (346-347).

Kopf, S., Haenselmann, T., Farin, D., Effelsberg, W. (2004). Automatic Generation of Summaries for the Web. *Proc. of IS&T/SPIE Electronic Imaging* (EI), Vol. 5307, pp. 417 – 428.

Kopf, S., Haenselmann, T., Effelsberg, W. (2005). Shape-based Posture and Gesture Recognition in Videos. *Proc. of IS&T/SPIE Electronic Imaging* (EI), Vol. 5682, pp. 114 – 124.

Kopf, S., Haenselmann, T., Kiess, J., Guthier, B., Effelsberg, W. (2011). Algorithms for video retargeting. *Multimedia Tools and Applications* (MTAP), Vol. 51 (2), pp. 819 – 861.

Kopf, S., Wilk, S., Effelsberg, W. (2012). Bringing Videos to Social Media. *Proc. of IEEE International Conference on Multimedia and Expo* (ICME), pp. 681 – 686.

Lampi, F., Kopf, S., Benz, M., Effelsberg, W. (2008a). A Virtual Camera Team for Lecture Recording. *IEEE MultiMedia Journal*, Vol. 15 (3), pp. 58 – 61.

Lampi, F., Kopf, S., Effelsberg, W. (2008b). Automatic Lecture Recording. *Proc. of the 16th ACM international conference on Multimedia* (MM), pp. 1103 – 1104,

Mentzel, W., & Flume, P. (2008). *Rhetoric* (in German), Planegg, Munich: Haufe publishing.

Nöllke, C. (2011). *Presentation* (in German), Freiburg: Haufe publishing.

Pabst-Weinschenk, M. (1995). *Talking in studies: a training program* (in German), Berlin: Cornelsen Scriptor.

Panger, G. (2012). Kinect in the kitchen: testing depth camera interactions in practical home environments, in: *CHI '12 Extended Abstracts on Human Factors in Computing Systems* (1985–1990).

Rahman, A. M., Saboune, J., & El Saddik, A. (2011). Motion-path based in car gesture control of the multimedia devices, in: *Proceedings of the first ACM international symposium on Design and analysis of intelligent vehicular networks and applications* (69–76).

Ren, Z., Yuan, J. Meng, J., & Zhang, Z. (2013). Robust part-based hand gesture recognition using Kinect sensor, *IEEE Transactions on Multimedia*, (Volume 15, Issue 5, 1110–1120).

Richter, S., Kühne, G., Schuster, O. (2001). Contour-based Classification of Video Objects. *Proc. of IS&T/SPIE Electronic Imaging* (EI), Vol. 4315, pp. 608 – 618.

Schön, D., Kopf, S., Schulz, S., Effelsberg, W. (2102a). Integrating a Lightweight Mobile Quiz on Mobile Devices into the Existing University Infrastructure. *World Conference on Educational Media and Technology* (EdMedia), pp. 1901-1907.

Schön, D., Kopf, S., Effelsberg, W. (2012b). A lightweight mobile quiz application with support for multimedia content. *International Conference on e-Learning and e-Technologies in Education* (ICEEE), pp.134-139.

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., & Blake, A. (2011). Real-time human pose recognition in parts from single depth images, in: *IEEE Conference on Computer Vision and Pattern Recognition* (1297–1304).

Shum, H., Ho, E., Jiang, Y., & Takagi, S. (2013). Real-time posture reconstruction for Microsoft Kinect, *IEEE Transactions on Cybernetics* (Volume 43, Issue 5, 1357–1369).

Wang, C., Liu, Z., & Chan, S. (2015). Superpixel-Based Hand Gesture Recognition With Kinect Depth Camera, *IEEE Transactions on Multimedia* (Volume 17, Issue 1, 29-39).

Wilk, S., Kopf, S., Effelsberg, W. (2013). Social Video: A Collaborative Video Annotation Environment to Support E-Learning. *Proc. of World Conference on Educational Multimedia, Hypermedia and Telecommunications* (EdMedia), pp. 1228-1237.

Yang, Z., Zicheng, L., & Hong, C. (2013). RGB-depth feature for 3D human activity recognition, *Communications, China* (Volume 10, Issue 7, 93–103).

Yao, Y., & Fu, Y. (2014). Contour Model-Based Hand-Gesture Recognition Using the Kinect Sensor, *IEEE Transactions on Circuits and Systems for Video Technology* (Volume 24, Issue 11, 1935-1944).