Enhancing the Communication Spectrum in Collaborative Virtual Environments

Edward Kim Christopher Moritz

Department of Computing Sciences
Villanova University, PA

Abstract. The importance of interpersonal and group communication has been studied and recognized for thousands of years. With recent technological advances, humans have enabled remote interaction through shared virtual spaces; however, research is still needed to develop methods for expressing many important non-verbal communication cues. Our work explores the methods for enhancing the communication spectrum in collaborative virtual environments. Our primary contribution is a machine learning framework that maps human facial data to avatars in the virtual world. We developed a synthetic training process to create labeled data to alleviate the burden of manual annotation. Additionally, we describe a collaborative virtual environment that can utilize both verbal and non-verbal cues for improved user communication and interaction. Finally, we present results demonstrating the success of our method in a sample collaborative scenario.

1 Introduction

Collaborative virtual environments (CVEs) are digital spaces where participants can collaborate and interact with one another. These are typically networked software platforms where the participants are represented as avatars in the virtual space. In different CVEs, users have vastly differing communication capabilities where verbal communication is the most dominant communication modality. However, only relying on verbal communication can be problematic. Research has shown that our words only account for 7% of our overall message [1], and in fact most of our communication spectrum resides in the non-verbal space. Some examples of non-verbal communication cues include facial expressions, gaze, tone of voice, and body language. Thus, for CVEs to truly emulate the entire range of human communication, the non-verbal cues need to be presented and expressed in the virtual world.

Our work explores a method of enhancing the communication spectrum in collaborative virtual environments. Our primary contribution is a framework for a flexible facial expression model that can map keypoint descriptors on a face to facial blendshape coefficients. Our model can easily be trained to map any number of keypoints to blendshapes using our described random face generator. Thus, the burden of labeling training data for alternative machine learning approaches is alleviated by our approach. Additionally, we describe the methodology and tools required to build a networked CVE that utilizes both verbal and non-verbal cues for virtual collaboration. Finally, we present the quantitative and qualitative results of our method on facial expression videos, a standard expression dataset, and demonstrate the non-verbal cues in a sample CVE scenario.

2 Background

Collaborative virtual environments have been researched for decades, but have been increasing in functionality and expressiveness in recent years. Fabri et al. [2] has explored the use of emotionally expressive agents to engender empathy amongst the users. Their work used the Facial Action Coding System (FACS) to express facial activity in avatars [3]. Action Units (AUs) that correspond to facial muscle motion were tweaked for expressive virtual agents. The impact of gaze and avatar realism was explored by Garau et al. [4]. Tanenbaum et al. [5] provides a thorough summary of the state of non-verbal communication in virtual worlds and states that the lack of non-verbal communication in virtual worlds leads to a state of confusion for users. Others have noted that more natural perception of each other (and of autonomous actors) increases their sense of being together, and thus the overall sense of shared presence in the environment [6].

As noted by the literature, there is abundant evidence demonstrating the importance of both verbal and non-verbal communication in CVEs. Towards this goal, researchers have branched out to other fields of computing for solutions. For facial expressiveness, a significant body of work from the computer vision community exists describing how to localize keypoints on the face [7–9]. Given these keypoints, one could map (or retarget) a human face expression to a virtual avatar. For facial keypoint retargeting, some have had success with facial rigging that requires manual tweaking [10] or with commercial software and blackbox packages [11, 12]. Alternative methods of facial retargeting can ease the burden on computer vision systems by leveraging hardware improvements i.e. RGB-D and depth sensors [13, 14]. Distinct from many of these methods, our work does not require manual tweaking, nor time consuming manual keypoint annotation for training. We can use simple webcams instead of RGB-D cameras for wide spread compatibility and integrate these verbal and non-verbal communication in CVEs.

3 Methodology

For our collaborative virtual environment, we are using the Unity3D[1] game engine. Unity3D provides the necessary tools and libraries for networking, facial

[1] http://www.unity3d.com

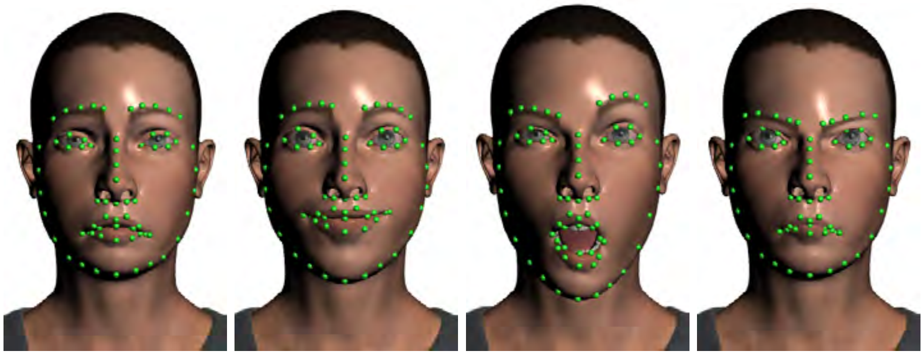


Fig. 1. Visualization of the 68 vertices tracked using a random face generation blendshape script. In order to train the mapping between the blendshape coefficients and the tracked facial points, we generate 2,000 random faces and project their vertex positions to x,y pixel locations.

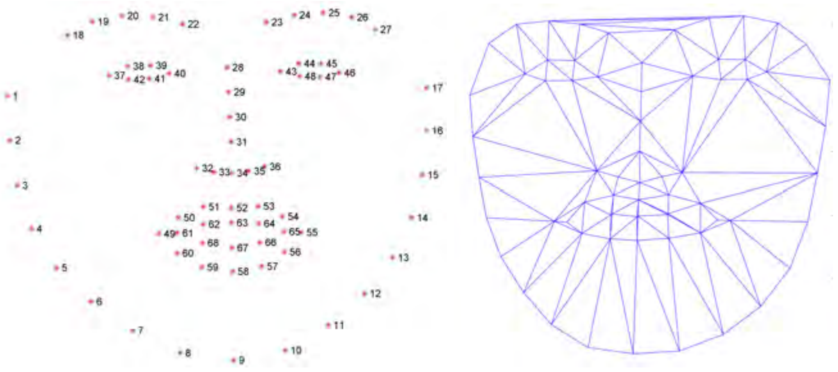
animation, and distribution. We obtain our 3D face models from Adobe Mixamo[2]. The models are auto-rigged with a skeleton and optionally can be enhanced with 50 facial blendshapes. These blendshapes can be manually tweaked to alter the facial movements of the 3D model. Some example blendshapes include a left and right eye blinks, cheek puff, outer eyebrow lower/raiser, mouth open, etc. The first task we describe is to build a facial expression model that can take arbitrary points on a human face and map their influence onto these 50 blendshapes. Then, we present our CVE system and describe the integration of both verbal and non-verbal cues.

3.1 Facial Expression Model

Synthetic Generation of Training Data - In a typical machine learning scenario, training a regression model requires a large amount of labeled data that can be expensive or time consuming to obtain. In the application of keypoint face localization, this is usually a very tedious task where a human annotator typically has to click on each point of the face. Existing facial position landmarks have been proposed in the literature (68 point markup from the Multi-PIE [15] dataset) which we adopt in our framework.

To alleviate the training burden, we developed a random face generator that generates a random face within a constrained range. Vertices of a face mesh that correspond to the standard 68 point markup are identified and tracked as facial blendshapes are randomly tweaked. The 3D vertex positions are projected into the 2D camera plane. We autonomously generate 2,000 faces for training a machine learning algorithm. Because the training data is synthetic, perturbations are added to the points for robustness and data augmentation purposes. Green

[2] http://www.mixamo.com



(a) Face Keypoint map (b) Delaunay Triangulation

Fig. 2. A visualization of the 68 keypoints used in our method and delaunay triangulation features extracted from our faces. The features correspond to the euclidean distances between points of the triangle mesh. These distances are normalized by the face width.

spheres have been placed on the corresponding mesh for visualization purposes, see Figure 1.

Training the Keypoint to Blendshape model - Our next task is to map these 68 face keypoints to avatar blendshape coefficients. This will enable any 68 point annotation on real human faces to transfer to a virtual character. We train this model using a two layer, feed forward neural network with hidden layer sigmoid activations and a linear output. We compute a delaunay triangulation of the keypoints of the face and compute the distance between triangle vertices. The input layer takes these 178 computed distances (see Figure 2 for an illustration of the distances used as features), passes it to a fully connected layer of 20 nodes, and forwards the data to a linear output layer. In total, we train 31 separate neural networks corresponding to different parts of the face. These neural networks include networks for major muscles of the face including eye control, mouth control, an upperface / brow network, lips, cheeks, nose, etc.

Human Face Keypoints from Video - Given a machine learning model that maps keypoints to blendshapes, we can map human expressions to avatars. After an evaluation of several facial keypoint computer vision methods, we chose the CLM [9] framework which utilizes a constrained local neural field for landmark detection. In our experiments, it was the most robust in video annotation with variable lighting conditions, and had a relatively fast processing time (< 0.3 seconds per frame on an Intel i7). For proper normalization, we create a common alignment for all the faces so we can more precisely detect the variations between expressions. We utilize the extended Cohn-Kanade (CK+) face database [16, 17] to compute a mean face shape across the entire dataset. Each individual face is aligned to the mean shape through an affine warp of the outer 27 keypoints of the face and 4 keypoints of the nose bridge. We did not use a full 68 keypoint warp, as it could distort the facial muscle activations.

3.2 Other Non-Verbal and Verbal Communication

Verbal Communication- The requirement of our CVE is to enable both verbal and non-verbal communication. Thus, we addressed the verbal communication through the standard high level API (HLAPI) multiplayer library for Unity. A user of our system can speak into a standard webcam microphone and their voice packets will be distributed to all other networked players in the CVE. The audio is played back as a 3D sound in the virtual world and an additional "speaking" indicator appears above the avatar's head when the system detects that they are talking.

Avatar Body Transform and Head IK- The user's head orientation is also a very important non-verbal cue that we wanted to incorporate in the CVE. The user can control their avatar's overall transformation using a keyboard and mouse. Since our avatar is fully rigged with facial blendshapes and skeleton, we can control where the avatar's head is pointing by controlling the head IK "lookAt" position. When a user moves their mouse around the screen to change their camera view, the avatar's head IK follows the camera's Z axis. This transform is then synchronized across the network. For a more immersive experience, we also enabled the Oculus Rift virtual reality headset and synchronized the sensors with the avatar transforms. The Oculus provides both positional and orientational data that is used to control the translation of the body and head IK of the avatar. We discuss some of the limitations of the Rift in our experiments and results section.

Body Motion Capture Scripts - Finally, we enabled a suite of motion capture animations through the Unity mecanim animation system. These are triggered animations that play when a user performs certain actions e.g. walking, talking, or can be triggered manually by the user using mapped keyboard actions. We plan to explore the Oculus Rift Touch when it becomes commercially available and plan to experiment with other technologies like the skeleton tracking system in the Microsoft Kinect.

4 Results

For our experiments and results, we present our data on training the keypoint to blendshape model, and visualize some of the results of the regression model on webcam video and on publically available expression datasets. We also demonstrate our integrated framework of verbal and non-verbal communication in a sample CVE.

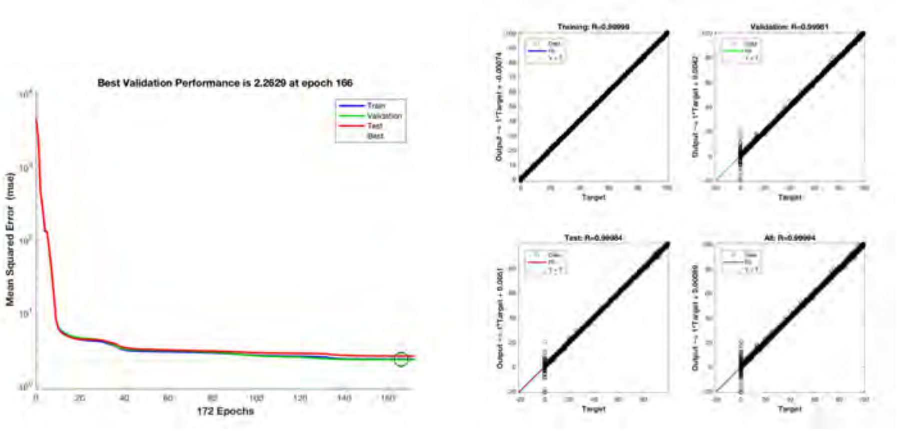


Fig. 3. Visualization of the training on our eye blendshape model. The model is trained with 70% of our data and validated and tested on remaining 30% of the data (15%, 15% split.) The training/validation/testing error is shown as well as a regression plot visualizing a linear fit to the data.

4.1 Training the Keypoint to Blendshape Model

As mentioned previously, we train 31 models to control the avatar's various blendshapes (eyes, mouth, upper facial muscles, lower facial muscles, etc.). Each model is trained using 70% of the labeled data, and the remaining 30% is used for testing and validation. The model uses the Levenberg-Marquardt algorithm and and L2 regularization to prevent overfitting. The training process runs for 1000 epochs or until a validation threshold is met. The training process is fairly quick, taking only 1-2 minutes to obtain each of the models. The training error and regression error can be seen in Figure 3, which is computed from the mean squared error between the output and targets. The test triangulation distance data is transformed to have zero mean and variance of 1.

4.2 Extended Cohn-Kanade (CK+)

The extended Cohn-Kanade (CK+) dataset is a fairly standard and popular dataset for expression recognition that consists of 593 recordings of 123 subjects. The image sequences vary between 10 and 60 frames, but only the last frame (peak expression) has been FACS coded. Every frame has also been manually annotated with 68 facial keypoints. The dataset is primarily used for obtaining an average face for which all other faces can be affinely transformed and normalized. Additionally, we can use the data for visualization of avatar emotion. The range of emotions expressed through this database covers a wide spectrum, including happiness, sadness, surprise, disgust, etc. We extract a peak expression frame from each video sequence, resulting in 593 frames. These keypoints are processed through our model and some of the visualizations can be seen in the last row of Figure 4.

4.3 Webcam Video

One of the primary goals of our work is to transfer expressions from standard webcam video to virtual characters. We captured video and utilized the CLM algorithm to analyze the frames of the video. Some results of our experiments can be seen in the first two rows of Figure 4. Unlike the single image snapshots of the CK+ dataset, with our video we performed temporal smoothing of the resulting blendshape coefficients using a moving average of 10 image frames. This reduced some of the "jittery" effects perceived when viewing the avatar's expressions in real-time.

Failure Cases There are number of failure cases in our testing that relate to the variability in the real world. The first failure case happens when our face keypoint detector fails to accurately localize the face. Most of the error occurs in the mouth region if the person greatly exaggerates an expression. This error propagates down the neural network and ultimately fails to register on the generated face. Another common failure situation are when the person expresses very slight muscle movement. Generally speaking, humans many be able to detect these subtle changes; however, our system has difficulty translating minor variances to the model.

4.4 Sample CVE scenario

We built a sample collaborative virtual environment scenario around the application of medical training and evaluation. Multiple users can join and interact with each other in a shared virtual operating room. A user can verbally communicate with everyone in the room using a standard microphone, and a "speech" indicator pops up over their head to draw the participants attention, see Figure 5(a). The audio is fully 3D, meaning that one can use stereo sound to localize the audio source in 3D space. Non-verbal cues in the CVE include playback of our facial keypoint blendshape model (current work in progress to make this run at 30 frames per second), and head joint IK to control the avatar's gaze direction. Pre-scripted body animations can be triggered by the user; however, hand tracking devices could easily be added for avatar hand IK motion. The Oculus Rift virtual reality headset has also been integrated, and the head orientation and motion mapped to the avatar's head, see Figure 5(b). Unfortunately, the current headset occludes the face and interferes with the facial keypoint algorithm for accurate facial expression transfer. Future work will look into building a machine learning face keypoint model that can support virtual reality headsets as demonstrated by Li et al. [18].

5 Conclusion

In summary, we present a framework for enhancing the communication spectrum in collaborative virtual environments. Our primary contribution is a machine learning framework that maps facial keypoint data to facial blendshape

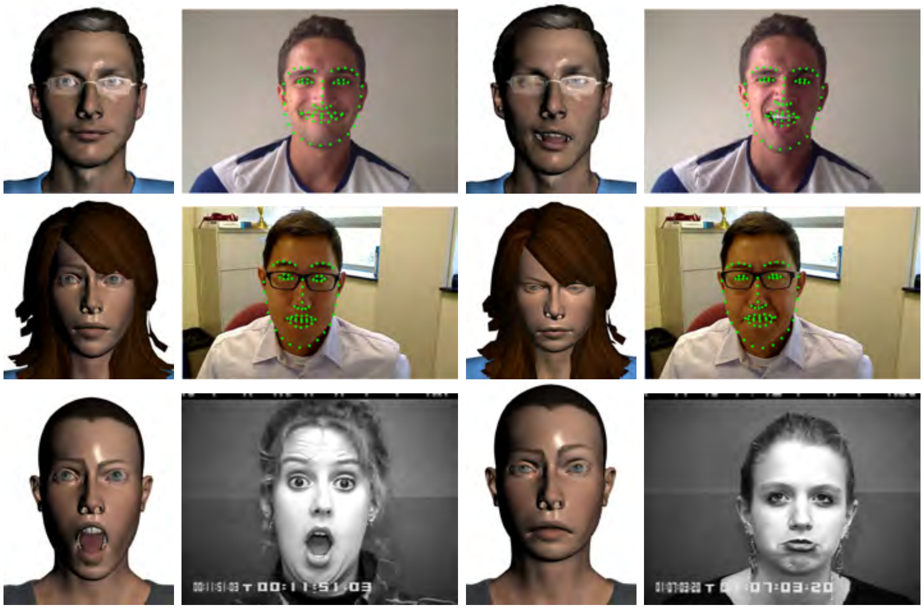
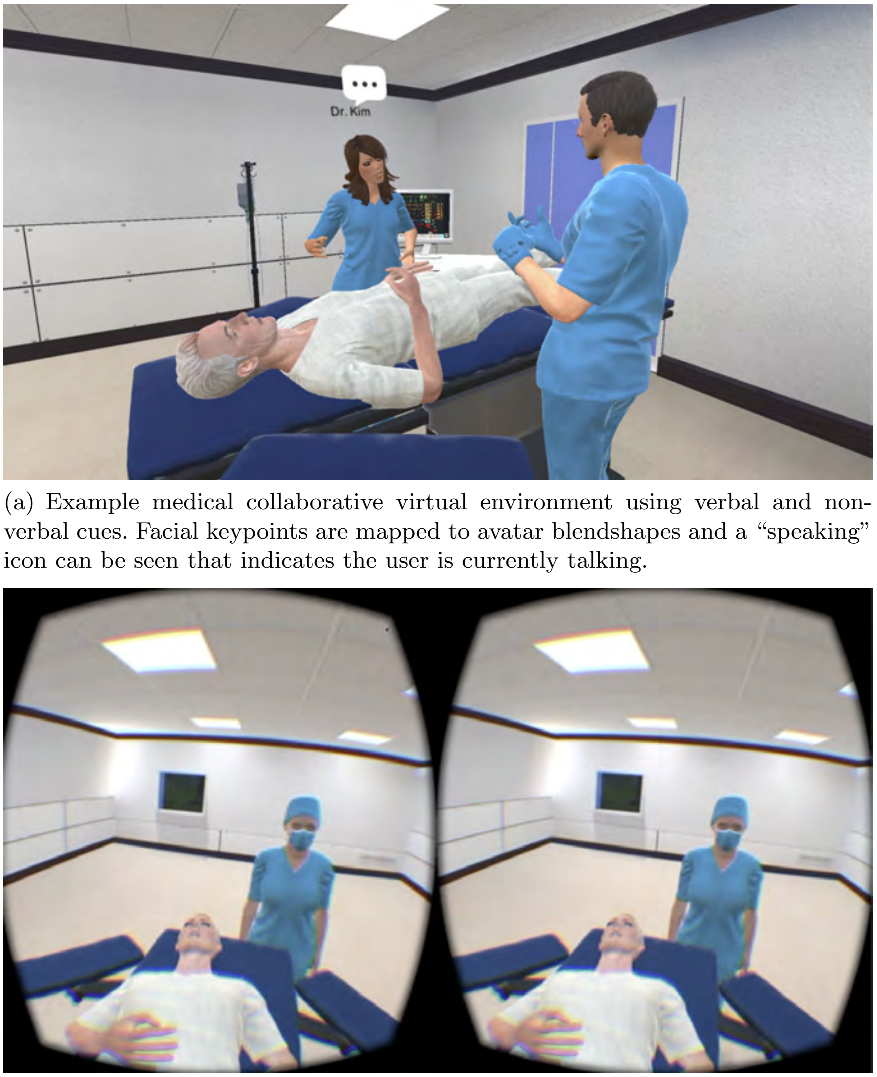


Fig. 4. Visualization of results of our model when mapping CLM facial keypoints to avatar blendshapes. The first two rows of the figure illustrate results on standard webcam video. The last row demonstrates the results on images from the CK+ dataset. The CK+ dataset provides manual keypoint coordinates so we did not need to perform any computer vision processes.

coefficients. We developed a synthetic training process to create labeled data to alleviate the burden of manual annotation. Additionally, we describe a CVE built that can utilize both verbal and non-verbal cues for improved user communication and interaction. Finally, we present both quantitative and qualitative results demonstrating the success of our method in a collaborative scenario.

References

1. Mehrabian, A.: Nonverbal communication. Transaction Publishers (1977)
2. Fabri, M., Moore, D.: The use of emotionally expressive avatars in collaborative virtual environments. Virtual Social Agents **88** (2005)
3. Ekman, P., Rosenberg, E.: What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS). Oxford University Press (1997)
4. Garau, M., Slater, M., Vinayagamoorthy, V., Brogni, A., Steed, A., Sasse, M.A.: The impact of avatar realism and eye gaze control on perceived quality of communication in a shared immersive virtual environment. In: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM (2003) 529–536
5. Tanenbaum, J., Seif El-Nasr, M., Nixon, M.: Nonverbal Communication in Virtual Worlds: Understanding and Designing Expressive Characters. ETC Press (2014)



(a) Example medical collaborative virtual environment using verbal and non-verbal cues. Facial keypoints are mapped to avatar blendshapes and a “speaking” icon can be seen that indicates the user is currently talking.

(b) Example of the visualization through virtual reality headsets. The avatar’s head IK is mimicking the orientation of the user.

Fig. 5. Example medical simulation using a collaborative virtual environment. The non-verbal cues are mapped to the virtual avatars which include facial expression blendshape data rendered from facial keypoint analysis, scripted motion capture body animation, and inverse kinematic head motion indicating where the user is looking.

6. Guye-Vuillème, A., Capin, T.K., Pandzic, S., Thalmann, N.M., Thalmann, D.: Nonverbal communication interface for collaborative virtual environments. Virtual Reality **4** (1999) 49–59

7. Cootes, T.F., Edwards, G.J., Taylor, C.J., et al.: Active appearance models. IEEE Transactions on pattern analysis and machine intelligence **23** (2001) 681–685
8. Zhu, X., Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 2879–2886
9. Baltrusaitis, T., Robinson, P., Morency, L.P.: Constrained local neural fields for robust facial landmark detection in the wild. In: International Conference on Computer Vision Workshops. (2013) 354–361
10. Sintel: Sintel face rig. `https://durian.blender.org/about/` (2016)
11. Mixamo: Adobe mixamo. `https://www.mixamo.com/faceplus` (2016)
12. Faceware: Faceware technology. `http://facewaretech.com/products/software/` (2016)
13. Realsense: Intel realsense technology. `https://software.intel.com/en-us/articles/realsense-overview` (2016)
14. Hsieh, P.L., Ma, C., Yu, J., Li, H.: Unconstrained realtime facial performance capture. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 1675–1683
15. Gross, R., Matthews, I., Cohn, J., Kanade, T., Baker, S.: Multi-pie. Image and Vision Computing **28** (2010) 807–813
16. Kanade, T., Cohn, J.F., Tian, Y.: Comprehensive database for facial expression analysis. In: IEEE Conference on Automatic Face and Gesture Recognition. (2000) 46–53
17. Lucey, P., Cohn, J., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In: Computer Vision and Pattern Recognition Workshops. (2010) 94–101
18. Li, H., Trutoiu, L., Olszewski, K., Wei, L., Trutna, T., Hsieh, P.L., Nicholls, A., Ma, C.: Facial performance sensing head-mounted display. ACM Transactions on Graphics (TOG) **34** (2015) 47