SIGGRAPH 2006

Course Notes:
Performance Driven Facial Animation

Parag Havaldar

Modeling a face and rendering it in a manner that appears realistic is a hard problem in itself, and remarkable progress to achieve realistic looking faces has been made from a modeling perspective [1, 6, 13, 15, 16, 2] as well as a rendering perspective [5, 11, 12]. At last years Siggraph 2005, the course of Digital Face Cloning described relevant material to this end. An even bigger problem is animating the digital face in a realistic and believable manner that stands up to close scrutiny, where even the slightest incorrectness in the animated performance becomes egregiously unacceptable. While good facial animation (stylized and realistic) can be attempted via traditional key frame techniques by skilled animators, it is complicated and often a time consuming task especially as the desired results approach realistic imagery. When an exact replica of an actor's performance is desired, many processes today work by tracking features on the actor face and using information derived from these tracked features to directly drive the digital character. These features, range from a few marker samples [3], curves or contours [15] on the face and even a deforming surface of the face [2, 16]. This may seem like a one stop process where the derived data of the performance of an act can be made to programmatically translate to animations on a digital CG face. On the contrary, given today's technologies in capture, retargeting and animation, this can turn out to be a rather involved process depending on the quality of data, the exactness/realness required in the final animation, facial calibration and often requires expertise of both artists (trackers, facial riggers, technical animators) and software technology to make the end product happen. Also, setting up a facial pipeline that involves many actors' performances captured simultaneously to ultimately produce hundreds of shots, with the need to embrace inputs and controls from artists/animators can be quite a challenge. This course documents attempts to explain some of the processes that we have understood and by which we have gained experience by working on Columbia's *Monster House* and other motion capture-reliant shows at Sony Pictures Imageworks.

The document is organized by first explaining general ideas on what constitutes a performance in section 1. Section 2 explains how facial performance is captured using motion capture technologies at Imageworks. The next section 3 explains the background research that forms the basis of our facial system at Imageworks – FACS, which was initially devised by Paul Eckman *et al.* Although FACS has been used widely in research and literature [7], at Sony Pictures Imageworks we have used it on motion captured facial data to drive character faces. The following sections 4, 5, 6 explain how motion captured facial data is processed, stabilized, cleaned and finally retargeted onto a digital face. Finally, we conclude with a motivating discussion that relates to artistic versus software problems in driving a digital face with a performance.

1 What Constitutes a Performance?

A performance, in most cases, is understood to be a visual capture of an actor face talking and emoting either individually or in a group with other actors. This is often done by capturing a video performance of the actor and using video frames either purely for reference by an animator or further processing them to extract point samples and even deforming 3D surfaces which are then used to retarget onto a digital character. There are of course a variety of technological hurdles to overcome before the 2D or 3D reconstructed data can be used – camera calibration, tracking points, and reconstructed 3D information. Apart from video media, other media types such as audio have also been used to capture a voice performance and drive digital faces. Most of the work here [4, 9] approximates the lip and mouth movement of lines of speech but does not have any explicit information that relates to other areas of the face such as brows, eyes and the overall emotion of the character. These have to be either implicitly derived or added on as a post process. Another medium of driving a digital face has been to use facial puppeteering, where a control device such as a cyber glove is used to drive a face. In such cases finger movements are retargeted on to the face.

While all these forms of capture to drive a face have yielded interesting results, the most common by far has been optical data that is used to reconstruct certain facial feature points that then are retargeted onto and drive a digital face.

2 Facial Motion Capture at Imageworks

Imageworks has had a fair amount of success in producing full-length CG animated movies based entirely on motion capture – both body and face. It began with *The Polar Express,* where the Imageworks Imagemotion™ facial and body motion capture acquisition, processing and animation pipeline was setup and successfully tried. The feedback from this experience helped to significantly enhance the pipeline for *Monster House* and now *Beowulf*. Motion capture has also been used to drive faces of digital characters in *Spider-Man 3*. Each productions needs are different, for example in *Monster House*, the motion capture system captured data of body and face together. The facial data had to be ultimately targeted onto characters whose faces were stylized and did not conform to the actual actor faces. For *Beowulf*, the requirements are more geared to producing realistic animation on characters that are supposed to look real and faces perform real. On *Spider-Man 3*, the facial motion captured data was acquired separately in a sit down position and the facial animation generated was blended in key framed body shots. The different types of systems used, the number of people simultaneously captured and facial only vs. face and body capture -- all result in varying data quality, creating many challenges to make data driven facial animation work well. At Imageworks, the artists and engineers are constantly involved to devise a universal system, which can adapt to these different production requirements.

On *Monster House*, face and body was captured simultaneously with two hundred cameras creating a capture volume of 20 feet x 20 feet x 16 feet (length, breadth, height). Eighty infrared markers were used on an actors face to capture the actor's performance. The data was captured and reconstructed in 3D using the multiple-camera system and post processed using, among a variety of tools, Imageworks proprietary Imagemotion technology which is adapted to capturing and processing motion data. On an average, there were about four actors in the volume, with a maximum of six at any given time.

In actual use, during any take, all the actors are made to stand apart in a standard T pose position – a position where the legs are together, hands are stretched out and the face is relaxed. This position is tremendously useful for a variety of search and standardization during the mocap data processing both for the body and face. Also, each take ends in all the actors returning to a standard T-pose in the volume. The T-pose is used by the facial pipeline in the normalization process to ensure that marker placements are made to correspond (as far as possible) to those on the day of the calibration (master T-pose). More on this is explained in section 4 and illustrated in Figure 1 below.

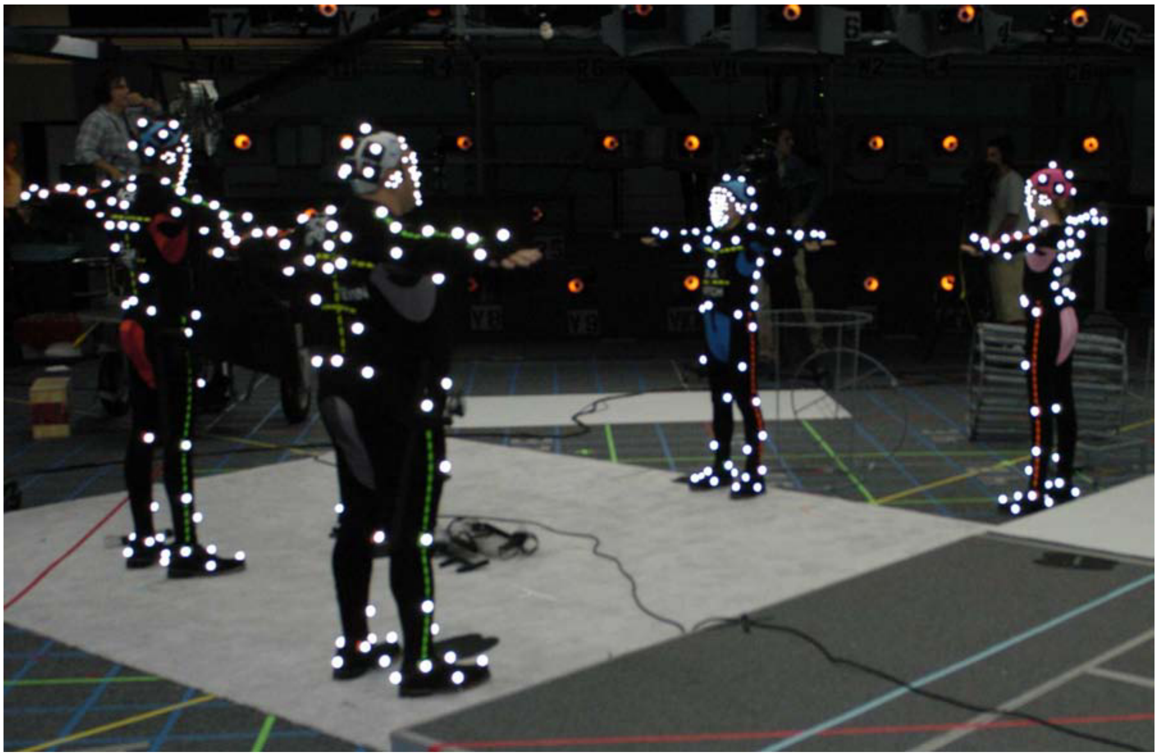


Figure 1. Motion capturing actors on "Monster House." Each actor starts the take and also ends the take in a standard T-pose position with a relaxed face. This neutral facial pose is used for normalizing the day-to-day changes in marker placements.

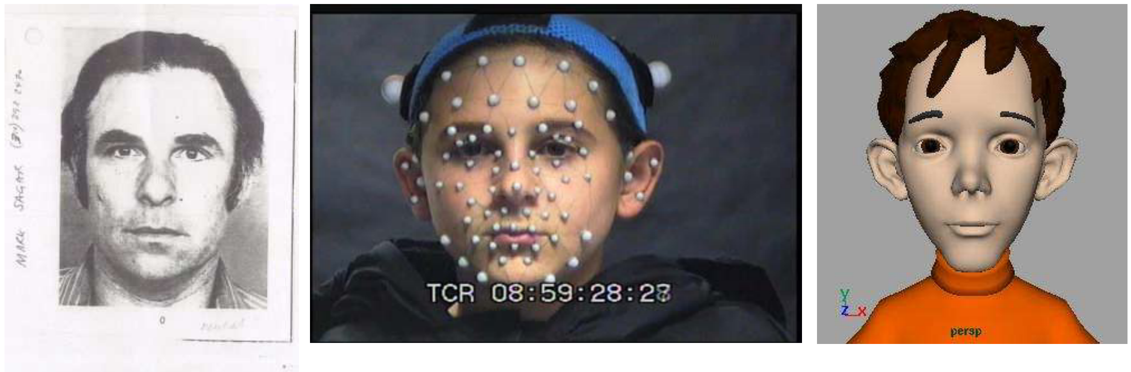
3 Facial Action Coding System (FACS)

The Facial Action Coding System or FACS was originally designed in the 1970s by Paul Eckman, Wallace Friesen and Joseph Hager [7]. Given that the face has muscles that work together in groups called Actions Units (AUs), FACS teaches how to understand when these action units are triggered and give a relative score to them. Although initially designed for psychologists and behavioral scientists to understand facial expressiveness and behavior, it has also been recently adapted in visual communication, teleconferencing, computer graphics and CG animation [8, 15].

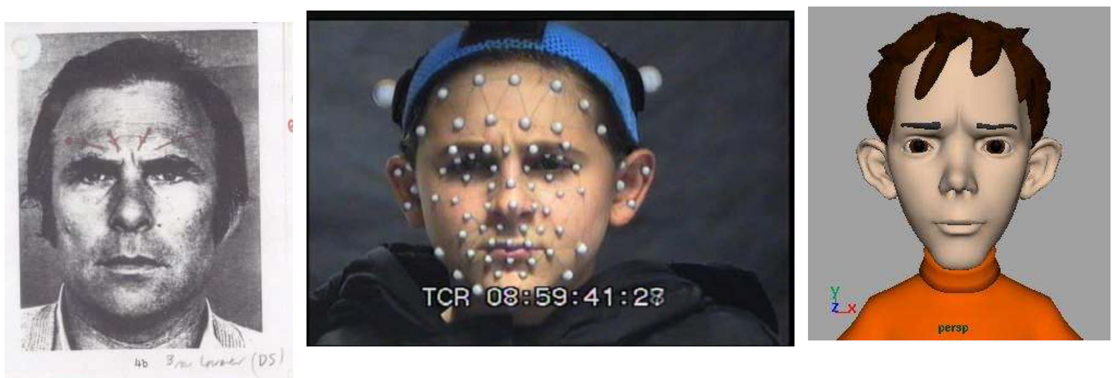
Eckman *et al.* [7], categorized facial expressions into 72 distinct Action Units. Each Action Unit represents muscular activity that produces momentary changes in facial appearance. This change in facial appearance of course varies from person to person depending on facial anatomy, e.g., bone structure, fatty deposits, wrinkles, shape of features etc. However, certain commonalities can be seen across people as these action units are triggered. The action units in FACS are based on the location on the face and the type of facial action involved. For example, the upper face has muscles that affect the eyebrows, forehead, and eyelids; the lower muscles around the mouth and lips form another group. Each of these muscles works in groups to form action units. The action units can further be broken down into left and right areas of the face, which can get triggered asymmetrically and independently of each other. In general, all the action units suggested by FACS give a broad basis for providing a good foundation for dynamic facial expressions that can be used in CG animation.

On *Monster House*, the facial system made use of FACS as a foundation to base the capture and retarget motion captured data on the characters' faces. Prior to acting, each actor went through a calibration phase where the actor is made to perform all the action units. The 3D reconstructed facial pose data that corresponded to one-action unit captures the extreme positions of how the actors perform a certain action. We used 64 poses, some of which were split into left and right positions along with 18 phoneme poses which described how actors articulated a phoneme. A complete list of these 64 facial expressions is provided in reference [7]. A few of the expressions are described below. In each case, the actual FACS reference, the actor's performance, and the retargeted expression on the character are shown.

Neutral Pose



Brow Lowerer Pose



Lip Corner Puller Pose

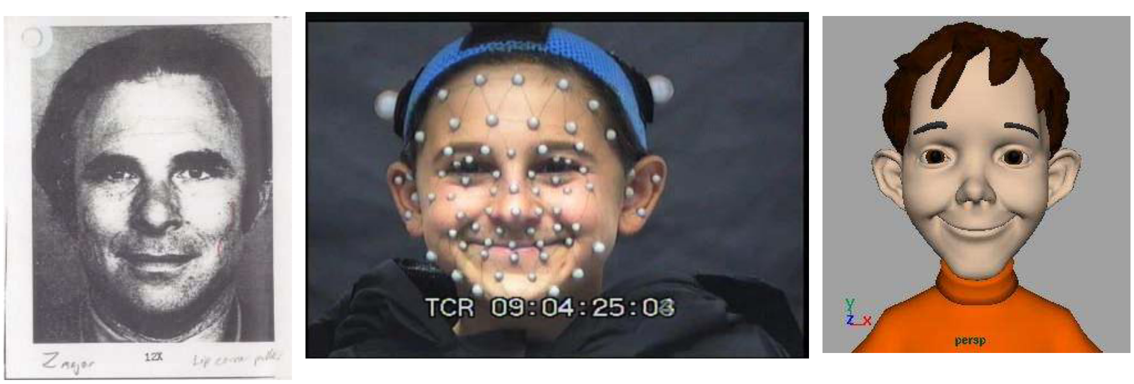


Figure 2. Example poses in a FACS based facial system. The reference action unit image is shown on the left, the actor's performance of the pose is shown in the middle and the artistic interpretation on the character is shown on the right.

Mouth Stretch Pose

Lip Stretcher Pose

Figure 3. More Example poses in a FACS based facial system. The mouth stretch and lip stretcher pose are shown. On "Monster House," the facial system used a total of 64 poses including 18 phoneme positions.

4 Data Capture and Processing

Data capture on *Monster House* was performed using an optical system with 200 infrared cameras capturing both body and face together. The capture volume was 20x20x16 feet and, on average, the performance of four actors was simultaneously captured. The number of facial markers used was 80. The capture system is a passive optical system that uses infrared cameras to capture infrared light reflected by the markers. The image from such cameras is a low entropy image consisting of mostly black areas where no infrared light was seen, with a few white dots whose size in the image varies depending on body/face marker, distance of the actor from the camera, occlusion caused by self and other actors, etc. The low entropy images have two main advantages:

- Due to low entropy, the cameras can capture and record images at higher definitions and at higher frame rates, typically at 60 Hz
- The 3D reconstruction problem triangulates points from multiple images at different viewpoints to get the marker location in space and needs to associate point correspondences automatically from the images. This correspondence problem in general is difficult to resolve, but is greatly improved by having only white dots to work with.

After 3D reconstruction, we now have (x,y,z) positions for markers in frames. However the data is bound to be noisy, does not have temporal associativity (i.e., consistent labeling) across all the frames, and may have gaps. A few typical data frames for the facial markers are shown in the Figure 4 below. These problems are sorted by using a learning based approach that learns both from a facial data model as well as the temporal quality of the data.

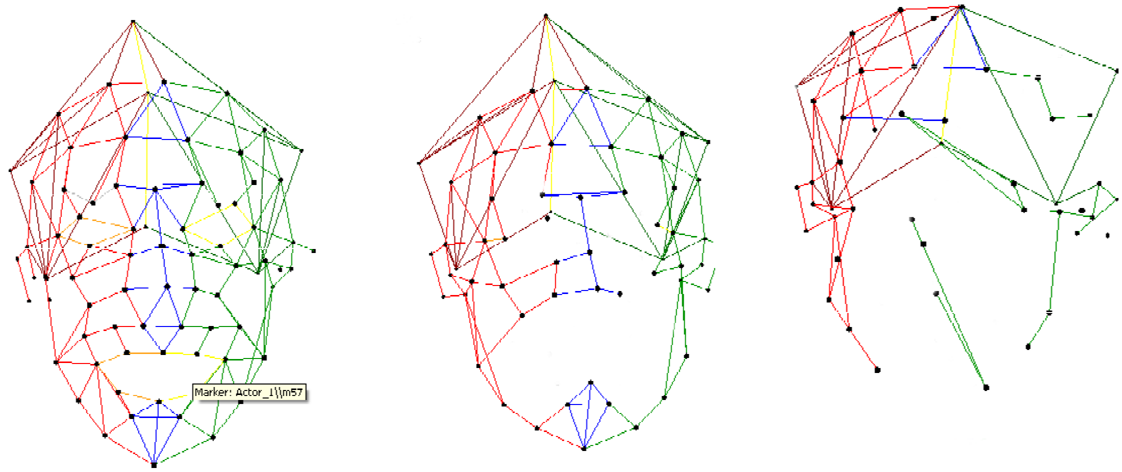


Figure 4. Facial data quality. On a large volume, multiple-person capture stage, the facial data capture quality varies depending on the number of people, self and inter-actor occlusions, etc. Good quality data is shown on the left, but at times this may significantly degrade as shown on the right.

4.1 Segmenting and Labeling Facial Data

The markers reconstructed for each frame have both body markers and face markers. Prior to any processing of facial data, the markers need to be labeled. Labelling is a process by which 3D reconstructed markers in all frames are consistently marked to have the same label. Labeling all markers (face and body) is normally based on trajectories can prove to be a very cumbersome and prone to error, especially with the number of markers visible in the volume and frequently required manual labor. We accomplished this task in a two step process by making use of the fact that body marker sizes are considerably larger than the facial markers. The first step involved tuning the 3D reconstruction so that only body markers are reconstructed (no facial markers) and consistently labeling the body markers using velocity-based constraints. Next, the 3D reconstruction can be tuned to acquire facial markers, which will also get body markers. The body markers can then be removed by using labeled data from the first step, resulting in only facial data.

4.2 Stabilization

During a performance, the actor is moving around in the volume resulting in the face markers being translated along with body, while the actor is speaking and emoting. However, for the purposes of retargeting the facial marker data onto the face, the facial data needs to be made stationary without the effect of body and head movement. The nullification of head rotations and body movement from the facial marker data is known as stabilization. Stabilization can be a tricky problem to solve because the face markers do not necessarily undergo a rigid transform to a standard position as the actor continues acting. There are rigid movements caused by head rotations and the actor's motion, but when the actor starts emoting and speaking, all the facial markers change positions away from their rigid predictions. Typically, to solve for the inverse transformation, a few stable point correspondences should be enough but it is not always easy to detect, on a frame-by-frame basis, which markers are stable and have only undergone a change due to the rigid transformation. The noise in the markers 3D reconstructed position also adds to the problem further.

To solve this problem, the Imageworks artists and engineers developed tools that solve the problem in a hierarchical manner by first doing a *global* or *gross stabilization* choosing selective markers, and further refining the output by *local* or *fine stabilization* by analyzing the marker movements relative to a facial surface model. The gross stabilization involves using markers that mostly do not move due to facial expressions – such as markers on the head, ears and the nose bone.

4.3 Facial Cleaning

Although the facial data has been stabilized, the data does contain missing markers due to occlusions and lack of visibility in the cameras, noise caused by errors in 3D reconstructions and occasional mislabeled markers. The cleaning and filtering process developed on *Monster House* made use of a learning system based on good facial model data that helped estimate missing markers, could remove noise and in general ensured the legality of all markers. The system was made scalable enough to handle wide ranges in facial expression as well as be tuned to ensure that the dynamics of the facial data.

These cleaning tools used the basic, underlying FACS theory of muscle groups to group markers together and organize them into groups of *muscles*. Muscle movements can be used to estimate probabilistically where the likely positions of missing markers should be. The predicted location is based spatially on the neighborhood points as well as temporally by studying the range of motion of the markers. It was our experience the probabilistic model and the marker muscle grouping had to be tuned to work for with each actor.

Once all the markers were predicted, standard frequency transforms were used to remove the noise in the data. However, care should be taken to understand the level of signal-to-noise ratio, which can drastically change over different frames for each marker. This is because high frequency content, which is normally categorized as noise, need not be so when the actor moves muscles and creates expression quickly.

4.4 Normalization

When capturing a long performance, such as a movie that spans over more than one day, actors need to remove motion captured markers and reapply them on a regular basis. Although appropriate steps can be taken to ensure that markers are placed at the same position on the face, it is common to have small differences between markers placements at every day positions. These differences can significantly affect the retargeting solutions described in the next section. Normalization is the process of adjusting the marker placements so that the differences in the positions do not compromise the extent of facial expression that the actor has performed and faithfully transfer that onto the character's face. Normalization is accomplished in two steps.

- Each take starts and ends with the actors performing a T-pose (see Figure 1). The T-pose of each actor in a specific take is oriented to the master T-pose of the actor, which was computed during the calibration phase mentioned above. This relies on certain relaxed landmark markers such as the corners of the eyes and mouth, which are expected to change very little from day to day. This alignment enables the computation of offset vectors for each marker.
- The offset vectors can now be applied to the take's T-position so that each marker in the T-pose is identical to the master T-pose. The offsets are then propagated through the actors performance so as to *normalize* all the frames.

5 Retargeting Motion Captured Data Using FACS

As seen in the previous section, the FACS theory gives a set of action units or poses which the authors [7] deem as a complete set for most facial expression. During a calibration phase, we capture mocap frames of the actor that relate to his facial expression corresponding to a FACS pose. Some of these poses are broken into left and right sides in order to capture the asymmetry that an actor face might undergo. Every incoming frame of the actor is then analyzed in the space of all these FACS captured frames. As such, these action unit-triggered poses correspond to facial basis vectors, and each one's activation needs to be computed for an incoming data frame. An illustration of this process is shown in Figure 5. The activations get transferred onto a digital face, which has been rigged using a facial muscle system. On *Monster House*, the facial poses on a character, which corresponded to FACS poses, were generated by an artist, requiring a facial rig. The facial setup on *Monster House* was based on Imageworks' proprietary character facial system (CFS). The system helps pull and nudge vertices on a 3D facial model so that the resulting deformations are consistent with what happens on a human face. It consists of four fascia layers each of which are blended together to create the final facial deformation.

- Muscle Layer– This is the main layer that consists of skull patches with muscled controls that deform the face. The muscle controls are activated by motion captured data.
- Jaw Layer – helps control the jaw movements.
- Volume Layer – helps add volume to the deformations occurring on the face. This helps to model wrinkles and other facial deformation, which can then be triggered by motion captured data.
- Artic Layer – This layer relates to the shape of the lips as they deform – in particular, the roll and volume of lips, essential when the character's lips thin out or pucker in facial expressions.

Also important are the eye positions. The facial rigging system starts by using a high resolution model of the face and filling in the various fascia layers which are driven by the motion captured data. The next step then is to define a way to map or retarget the motion captured data onto the face. This is accomplished by analyzing an incoming mocap frame in the space of all the facial basis vectors as captured using the FACS theory. The weighted values give the relative percentage by which the FACS action units or poses are triggered. These weights are computed using mathematical linear and nonlinear optimization techniques. However, we have found that the weights obtained by such rigorous analysis might not always correspond to good aesthetic deformations on the characters' face and sometimes require an artist's or animator's feedback to tune the system to perform as desired.

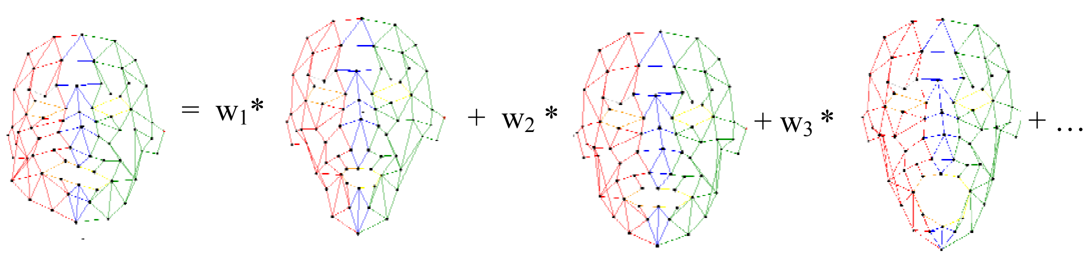


Figure 5. Computing w_1, w_2 ... w_n gives the activation of each FACS action unit. The computation of these weights needs mathematical linear / non-linear optimization techniques.

The output of the whole system is shown, for example, in the *Monster House* still images below.

6 Multidimensional Tuning

Although the facial retargeting achieved by various mathematical optimization processes may seem correct mathematically, they do not always necessarily conform to the requirements of a finalized animation shots. There may be a variety of reasons for why this happens:

- Actors not performing wholly according to the actors calibration
- Retargeting issues arising from mapping mathematically correct marker data to an artistically designed face
- Facial model not conforming to an actor's face
- Marker placements differ from day to day
- Attempting to produce an animation contrary to what the actor performed – either the expression is not there in the motion captured data or needs to be exaggerated

Most of these problems can be corrected by a tuning system, which recalibrates the facial [bais??] library based on feedback from an animator. At Imageworks, the artists and software engineers developed a "Multidimensional Tuning System," which takes an artist's input to reduce the effects of incorrect mathematical solves. In this system, post a FACS solve, the animator adjusts a few frames (typically five to ten, among the many thousands) to "look correct." This is accomplished by modifying the weights of poses on a "few" culprit frames that have resulted from a FACS solve. The tuning system exports this changed data, analyzes it and creates a more optimized FACS library. The new library generated is based on the marker range of motion as well as the changed weighting and uses non-linear mathematical optimization tools. This changed library when used in a new solve now attempts to hit the weighting defined by the animator on the few tuned frames at the chosen poses *and also* incorporate this change programmatically on to various other frames causing the whole animation to look better.

An Example of the tuned outputs on the facial library is shown in Figure 6 below

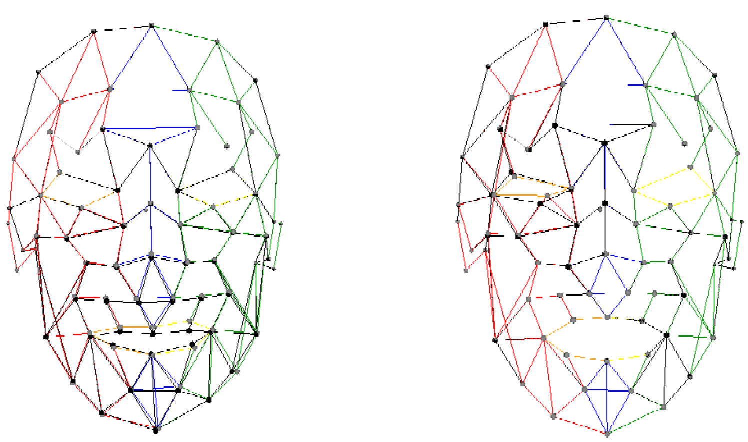


Figure 6. The images above show FACS poses overlaid before and after the tuning phase. The left image shows the lip shut phoneme position overlaid before and after, while the right image shows the same for the lid tightener pose.

In Figure 6, one can see that the new marker positions (in black) have been adjusted to an optimized location based on the animators corrected weighting values over a few tuned frames. This change is shown on two poses but occurs on more depending on the animator's input. Furthermore, the changes are done in such a way, when the FACS system resolves the solution with the new "tuned FACS matrix," not only does the solve hit close to the weights on the tuned frames, but also on all other frames creating a more correct animation. The effect of this on the retargeting solution on a character is shown Figure 7 and Figure 8.

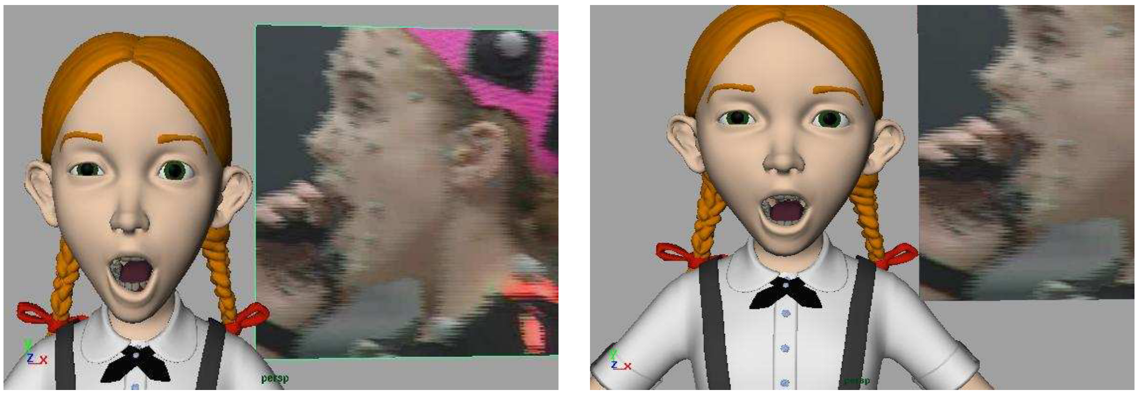


Figure 7. The left image above shows a solved frame with the original FACS calibrated matrix, the right one shows the solved frame with the tuned matrix. The effect of the tuned right lid tightener pose is evident.

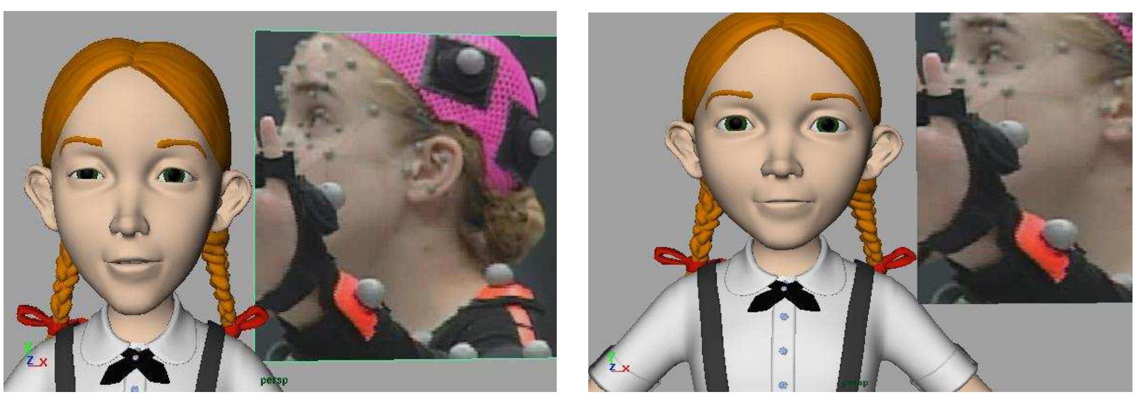
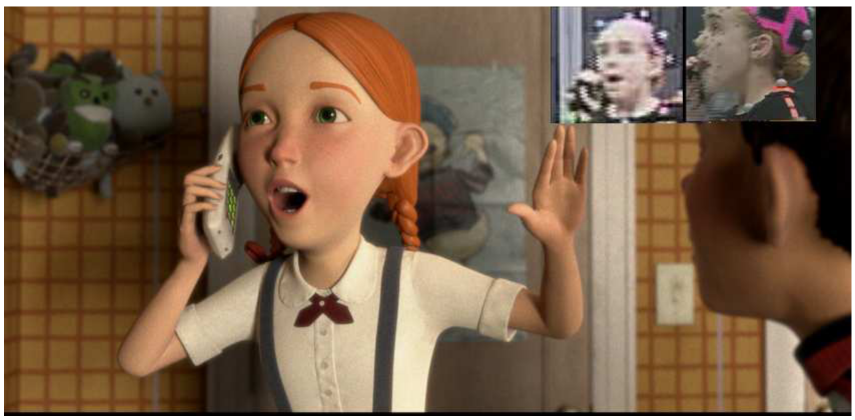


*Figure 8. The left image above shows a solved frame with the original FACS calibrated matrix, the right one shows the solved frame with the tuned matrix. The actor is saying the beginning of the word "**pl**ease." The original matrix solve does not show the lips closed to say the first syllable while the tuned matrix solve does.*


ND 12:37:02:23
52B-1B
D
T1


15:59:28:00
C

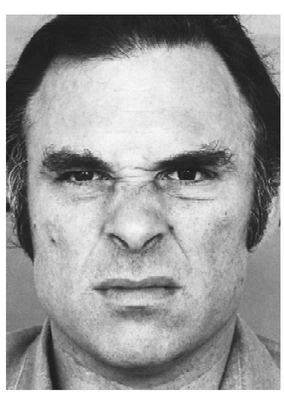


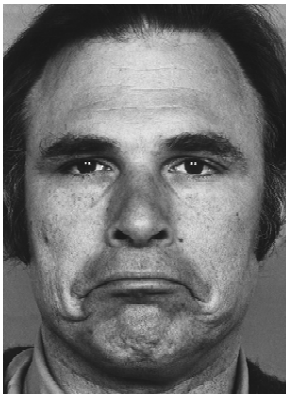
7 Discussion

We have attempted to describe the overall foundations of representation and analysis. The whole process is a combination of artistic input and mathematical processes to model a face and have data drive it via retargeting solutions. Here are a few interesting questions which should provoke good discussion.

- The retargeting solutions though mathematically correct, need not always work aesthetically where an animator's feedback is crucial and final touch ups necessary. How do programmers design tools that encapsulate the fidelity achieved by motion capture data, as well as, allow animators to massage or add on more artistic content additively – ***in such a manner that the two do not conflict***, and consistency is preserved temporarily across a shot(s) and for the entire production? To that end, the mathematical approaches in literature show how to get exact solves in some space (PCA, retargetted blend shapes, etc.), but having user assisted guidance to go towards mathematical minimizations in optimization that make sense aesthetically is not easy.

- Acting in a large volume with body and face, with many actors is what a director and a true performance capture system really desires. By being in the midst of the actual people, an actor is more in sync with his/her role and thereby emotes performances well -- this when compared to a sit down single scenario in front of a system. However, the data capture technologies from both cases yield substantial differences in fidelity – large volume captures with many people are not as stable as a sit down, one person capture. What course should be taken during solving, retargeting and animation such that the cost associated with data processing, clean up and the final integration and animation is reduced?
- Is facial rigging necessary? What does a rig get you in terms of being able to be driven by performance captured data sets. Rigging a face can be an expensive setup increasing production costs, but when and how does rigging solve these problems.
- Retargeting is not always an easy problem to solve, especially when the actor's facial feature and geometry do not coincide with the 3D facial model and rig – for example in case of a stylized character. How do we make universal systems in which the retargeting works effectively between real faces as well as stylized, humanoid characters that may also need exaggerated cartoon like movements?
- Are there any specific "rules" that you can use, whether mathematically founded, heuristically arrived at or even aesthetically pursued for when a person talks vs. muscle and skin movements on the rest of the face? To that end, how do you formalize a model for facial dynamics that are evident in small/large vibrations, jerks intonations when a persons face undergoes fast motion while acting an emoting e.g., moving head quickly from side to side, talking while running?
- Most performance driven techniques for facial animation including ours use visual optical data – video, infrared imagery, etc. What do other performance captures media give you and how can one use them more effectively to drive facial animations. There is various literature on usage of audio [4, 9], but how does a system use both?
- Wrinkles can have a lot to do with facial expression. How do you model wrinkles, and how does a programmatic system trigger their appearance and the extent to which they should be present in an expression? Do you need to physically model them in the 3D facial model, can their effect be achieved through shading and bump maps during the rendering process? A few wrinkles are shown below.

- Every CG face is modeled, textured, animated and rendered. Texturing and rendering are indeed necessary to make a face look real. But in order for the performance of an animated face to look real, to what extent does rendering help in getting close to a real performance?

8 Acknowledgments

All large production projects require artists, animators, integrators, software engineers and management. Cooperative work ultimately manifests in the final product on screen. Here is a list of people involved on the project in Monster House and other related shows that helped create this document.

Management – *George Joblove, Bill Villareal, Brian Keeney, Leslie Picardo, Jerome Chen, Jay Redd*
Motion Capture and Tracking – *Demian Gordon, Dennis J Huack, Darin Velarde, Nancy Ong*
Motion Capture Integration – *Albert Hastings, David Bennett, Josh Ochoa, Ron Fisher, Brian Doman*
Facial Rigging – *Brian Thompson, Michael Laubach, Rick Grandy, JJ Blumenkranz, Arthur Gregory*
Animation – *Troy Saliba, T. Daniel Hofstedt, Kenn Mcdonald, Remington Scott*
Training – *Sande Scoredos*
Marketing – *Carlye Archibeque*

9 References

1. V. Blanz, C. Basso, T. Poggio, and T. Vetter. Reanimating faces in images and video. In *Proc. Of Eurographics*, 2003.
2. George Borshukov, Dan Piponi, Oystein Larsen, J. P. Lewis, and Christina Tempelaar Lietz. Universal capture: image-based facial animation for ”the matrix reloaded”. In *Proceedings of SIGGRAPH Conference on Sketches & applications*. ACM Press, 2003.
3. E. Chuang and C. Bregler. Performance driven facial animation using blendshape interpolation. *CSTR- 2002-02, Department of Computer Science, Stanford University*, 2002.
4. Cosker, D.P., Marshall, A.D., Rosin, P.L., Hicks, Y.A., Speech-driven facial animation using a hierarchical model, *VISP(151), No. 4,* August 2004, pp. 314-321
5. Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *SIGGRAPH 2000 Conference Proceedings*, pages 35–42. ACM SIGGRAPH, July 2000.
6. Eisert, P., and Girod, B. 1997. Model-based facial expression parameters from image sequences. In *Proceedings of the IEEE International Conference on ImageProcessing* (ICIP-97), 418-421.
7. P. Ekman and W.V. Friesen, *Manual for the facial action coding system,* Consulting Psychologists Press, Palo Alto, 1977.
8. I. A. Essa and A. P. Pentland Facial expression recognition using a dynamic model and motion energy. *Proc. IEEE Int’l Conference on Computer Vision*, pages 360–367, 1995.
9. Theobald, B.J., Kruse, S.M., Bangham, J.A., Cawley, G.C., Towards a low bandwidth talking face using appearance models, *IVC(21), No. 12-13*, December 2003, pp. 1117-1124.
10. Tim Hawkins, Andreas Wenger, Chris Tchou, Andrew Gardner, Fredrik G¨oransson, and Paul Debevec. Animatable facial reflectance fields. In *Rendering Techniques 2004: 15th Eurographics Workshop on Rendering*, pages 309–320, June 2004.
11. Jensen H. W., Marshiner, S., Levoy, M., and Hanrahan, P. 2001. A practical model for subsurface light transport. *Proceedings of SIGGRAPH ’2001*, 511–518.
12. Jensen H. W. J Buhler, A Rapid Hierarchical Rendering Technique for translucent materials. *In Proceedings of SIGGRAPH 2005.*
13. Jun Yong Noh and Ulrich Neumann. Expression cloning. In *Proceedings of ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 277–288, August 2001.
14. Mark Sagar, Reflectance Field Rendering of Human Faces for “Spiderman 2”. SIGGRAPH 2004.
15. D. Terzopoulos and K. Waters. Techniques for realistic facial modeling and animation. In Nadia Magnenat Thalmann and Daniel Thalmann, editors, *Computer Animation 91*, pages 59–74. Springer-Verlag, Tokyo, 1991.
16. Li Zhang, Noah Snavely, Brian Curless, and Steven M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *ACM Trans. Graph.*, 23(3):548–558, 2004.