FAST-FACS: A Computer-Assisted System to Increase Speed and Reliability of Manual FACS Coding

Fernando De la Torre (1), Tomas Simon (1), Zara Ambadar (2), and Jeffrey F. Cohn (2)
1. Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA
2. University of Pittsburgh, Pittsburgh, PA 15260, USA

Abstract. FACS (Facial Action Coding System) coding is the state of the art in manual measurement of facial actions. FACS coding, however, is labor intensive and difficult to standardize. A goal of automated FACS coding is to eliminate the need for manual coding and realize automatic recognition and analysis of facial actions. Success of this effort depends on access to reliably coded corpora; however, manual FACS coding remains expensive and slow. This paper proposes Fast-FACS, a computer vision aided system that improves speed and reliability of FACS coding. Three are the main novelties of the system: (1) to the best of our knowledge, this is the first paper to predict onsets and offsets from peaks, (2) use Active Appearance Models for computer assisted FACS coding, (3) learn an optimal metric to predict onsets and offsets from peaks. The system was tested in the RU-FACS database, which consists of natural facial behavior during a two-person interview. Fast-FACS reduced manual coding time by nearly 50% and demonstrated strong concurrent validity with manual FACS coding.

Keywords: Facial Action Coding System, Action Unit Recognition

1 Introduction

FACS (Facial Action Coding System: [1]) coding is the state of the art in manual measurement of facial action. FACS coding, however, is labor intensive and difficult to standardize across coders. A goal of automated FACS coding [2–4] is to eliminate the need for manual coding and realize automatic recognition and analysis of facial actions.Success of this effort depends on access to *reliably coded corpora* of FACS-coded images from well-chosen observational scenarios. Completing the necessary FACS coding for training and testing algorithms has been a rate-limiter. Manual FACS coding remains expensive and slow.

The inefficiency of current approaches for FACS coding is not inherent to FACS but to the failure to make use of technology to make coders more productive. This paper proposes an hybrid system, Fast-FACS, that combines automated facial image processing with manual coding to increase the speed and reliability of FACS coding. Figure 1 shows the main idea of the paper. The specific aims are to: (1) Reduce time and effort required for manual FACS coding by using novel computer vision and machine learning techniques. (2) Increase reliability of FACS coding by increasing the internal consistency of manual FACS coding. (3) Develop an intuitive graphical user interface that is comparable to commercially available packages in easy of use, while enabling fast reliable coding.

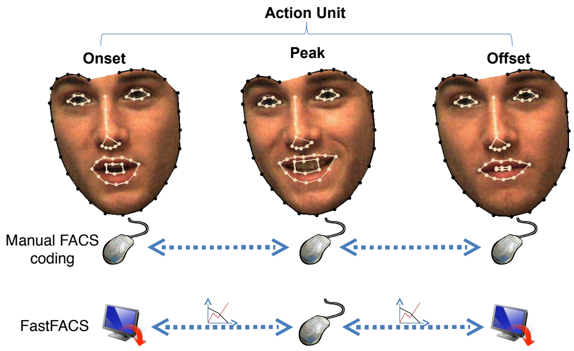


Fig. 1: FACS coding typically involves frame-by-frame inspection of the video, paying close attention to subtle cues such as wrinkles, bulges, and furrows. Left to right, evolution of an AU 12 (involved in smiling), from onset, peak, to offset. Using FastFACS only the peak needs to be labeled and the onset/offset are estimated automatically.

2 Previous work

2.1 Facial Action Coding System (FACS)

FACS [1] is a comprehensive, anatomically-based system for measuring nearly all visually discernible facial movement. FACS describes facial activity on the basis of 44 unique action units (AUs), as well as several categories of head and eye positions and movements. Facial movement is thus described in terms of constituent components, or AUs. FACS is recognized as the most comprehensive and objective means for measuring facial movement currently available, and it has become the standard for facial measurement in behavioral research [5].

Human-observer-based methods like FACS are time consuming to learn and use, and they are difficult to standardize, especially across laboratories and over time. A goal of automated FACS coding [2–4] is to eliminate the need for manual coding and realize automatic recognition and analysis of facial actions. However, the success of this effort depends on access to reliably coded corpora of FACS-coded images from well-chosen observational scenarios, which entails extensive need for manual FACS-coding.

Currently, FACS coders typically proceed in either single or multiple passes through the video. When a single-pass procedure is used, they view the video and code the occurrences of all target AU in each frame. FACS coders view video at both regular video rate and in slow motion to detect often subtle changes in facial features, such as wrinkling of facial skin, that indicate the occurrence, timing, and intensity of facial AUs. AU intensity is coded on a 5-point ordinal intensity scale (which provides a metric for the degree of muscular contraction) from trace to maximal intensity. FACS scoring produces a list of AUs, their intensity, and the video frames or times at which each began (i.e. onset), peaked (highest intensity observed), and ended (i.e., offset). Fig. 1 shows an example of how to code onset, peak and offset of AU12, which raises the lip corners obliquely. To date, manual FACS coding remains slow, and achieving and maintaining inter-coder reliability is challenging.

2.2 Automatic FACS segmentation and recognition from video

Advances in computer vision over the past decades have yielded advances toward the goal of automatic FACS. That is, to eliminate the need for manual coding and realize automatic recognition and analysis of facial actions.

Two main streams on automatic analysis of facial expression consider emotion-specified expressions (e.g., happy or sad) and anatomically based facial actions (e.g., FACS). The most relevant work is the one that addresses the temporal segmentation of AUs into onset, offset, and peak. Pantic and Pantras [4] used a rule-based method to separate onset, apex and offset. Valstar and Pantic [6] combined Hidden Markov Models and Support Vector Machines to model the temporal dynamics of facial actions. They considered the onset, apex, and offset frames as different classes. Accuracy was measured as precision-recall in these classes. These approaches all used supervised learning with the goal of fully automated expression or AU coding.

More recently, two groups have proposed hybrid systems that make use of more unsupervised learning techniques to augment manual coding of AUs. Zhang et al. [7] proposed an active learning approach to improve speed and accuracy in AU labeling. In their approach, a sequence is labeled with an automatic system, and a user then is asked to label the frames that are considered ambiguous by the system. De la Torre et al. [8] proposed an unsupervised algorithm to segment facial behavior into AUs, an approach that achieved concurrent validity with manual FACS coding. Subsequently, found that this unsupervised approach could achieve fast, accurate, robust coding of AU onsets and offsets when coupled with manual coding of AU peaks.

3 Fast-FACS

This section describes our Fast-FACS system, that uses advances in computer vision and machine learning to increase the efficiency and reliability of FACS coding.

3.1 Active Appearance Tracking

There exist a variety of methods for facial feature tracking. Over the last decade, appearance models have become increasingly prominent in computer vision and graphics. Parameterized Appearance Models (PAMs) have been proven useful for alignment, detection, tracking, and face synthesis [9–11]. In particular, Active Appearance Models (AAMs) have proven an excellent tool for detecting and aligning facial features. AAMs [9, 11, 10] typically fit their shape and appearance components to an image through a gradient descent, although other optimization approaches have been employed with similar results. Figure 1 shows how a person-dependent AAM [11, 9] is able to track the facial features during a segment that includes smiling (AU12). The person-dependent AAM is build with few samples (typically 15-20) of one person containing expression, pose and illumination changes in the video sequence. The AAM is composed of 66 landmarks that deform to fit perturbations in facial features. To the best of our knowledge, the work described here is the first to use the results of AAMs in a hybrid system to improve the speed and reliability of FACS coding. The hybrid system augments the skill of highly trained FACS coders with computer vision and machine learning based video editing and estimation of AU onsets and offsets.

3.2 Peak, Onset and Offset Coding

In the first step of Fast-FACS, the user annotates the peak of a facial action. The system then automatically determines the remaining boundaries of the event, that is, the onset and offset (extent) of the AU. The estimation of the position of the onset and offset of a given event peak is based on a similarity measure defined on features derived from the AAM mesh of the tracked face and on the expected distribution of onset and offset durations (for a given AU) derived from a database of manually coded AUs.

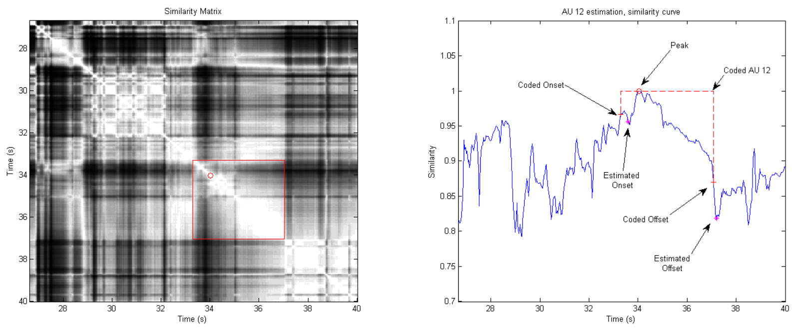


Fig. 2: Left) Similarity matrix for a video segment. The red rectangle denotes a specific AU 12 instance, as coded manually. The red circle marks the user-labeled peak. Observe that the AU defines a region "bounded" by sharp edges in the similarity matrix. Right) Similarity curve for the marked peak (ie. $\mathbf{K}_{peak,j}$ for all j in a neighborhood). Note how the estimated onset and offset snap to local minima on the similarity curve.

We construct a symmetric affinity matrix $\mathbf{K} \in \Re^{n \times n}$, where each entry $k_{ij} \in [0, 1]$ represents the similarity between frames i and j, and n denotes the number of frames [8]. This similarity measure will be used to decide where best to partition the AU into onset, peak and offset sections.

To compute the similarity measure (a qualitative distance from the peak frame), k_{ij}, we use the 66 shape landmarks from the tracking. The particular distance measure will be addressed in section 3.3. The description of the feature extraction process follows: The AAM mesh is first interpolated to a finer resolution using B-Spline fitting in the region of interest (upper or lower face). The resulting mesh from frame i is aligned with respect to frame j using an affine transform intended to remove the rigid movement of the head while retaining the elastic deformations of facial actions. Once both frames are commonly referenced, the landmarks are stacked into vectors $\mathbf{f}_i$ and $\mathbf{f}_j$, and $k_{ij} = e^{\frac{-d(\mathbf{f}_i, \mathbf{f}_j)}{2\sigma^2}}$ where $d(\mathbf{f}_i, \mathbf{f}_j)$ measures distance.

Figure 2 shows the similarity curve for a video segment. The similarity measure is robust to changes in pose (rigid motion) as well as to changes in facial expression that do not affect the particular AU under study (non-rigid motion). Additionally, the measure need to be invariant with respect to each AU class. The distance between frames is computed with the Mahalanobis distance $d(\mathbf{f}_i, \mathbf{f}_j) = (\mathbf{f}_i - \mathbf{f}_j)^T \mathbf{A} (\mathbf{f}_i - \mathbf{f}_j)$. Next section describes a metric learning algorithm to learn $\mathbf{A}$.

3.3 Learning a metric for onset/offset detection

This section describes the procedure to learn a metric [12] for onset and offset estimation. Let d features of each frame i be stacked into a vector, $\mathbf{f}_i \in \Re^{d\times 1}$. $\mathbf{f}_i^p$ denotes a frame i within a close neighborhood of an AU peak frame, $\mathbf{f}_p$. Let $\mathbf{f}_k^{o(p)}$ denote a frame at or beyond the onset (or offset) of the same AU. The metric learning optimizes:

$$\min_{\mathbf{A}} \sum_{i,p} (\mathbf{f}_i^p - \mathbf{f}_p)^T \mathbf{A} (\mathbf{f}_i^p - \mathbf{f}_p) + C \sum_{k,p} \xi_{k,p}$$
$$s.t.\ \sqrt{(\mathbf{f}_k^{o(p)} - \mathbf{f}_p)^T \mathbf{A} (\mathbf{f}_k^{o(p)} - \mathbf{f}_p)} > th - \xi_{k,p} \ \ \forall k,p \quad \mathbf{A} \succeq 0,\ \xi_{k,p} \geq 0 \qquad (1)$$

where $\mathbf{A} \in \Re^{n\times n}$ is a symmetric positive semi-definite matrix that minimizes the distance between frames neighboring the peak while ensuring that the distance between the peak and those frames at or beyond the onset and offset is greater than a given threshold th. Slack variables $\xi_{k,p}$ ensure that the constraints can be satisfied, while the parameter C adjusts the importance of the constraints in the minimization.

Eq. (1) can be solved with SDP approaches and we used the `CVX` package [13]. Restricting $\mathbf{A}$ to be diagonal is equivalent to individually weighting the features. While a full matrix could be used, in our experience diagonal and full matrices provide comparable performance, and we used this strategy for the experimental results.

3.4 Graphical user interface (GUI)

There exist several commercial packages for manual event coding. These systems are general purpose and must be adapted for FACS coding. They are proprietary systems, and therefore cannot be easily if at all modified to accommodate user developed modules, such as automatic onset and offset detection. We have developed a GUI specifically for FACS coding with the goal of creating an open-source framework that makes it possible to add new features (such as onset and offset detection) in order to improve the speed and reliability of FACS coding.

Fig. 3 shows the GUI for Fast-FACS. As described above, the coder goes through the video and manually codes the peaks that they detect, assigning an AU identifier and related features of the event (intensity and laterality), as well as comments about the peak to indicate whether it is gradual, ambiguous or an instance of multiple peaks. Annotation other than labeling of the peaks is for the user's reference and not used in onset or offset estimation. Once the peaks have been labeled, the onset and the offset are automatically detected and the resulting events made available for the user's inspection.

For FACS coders, it is usually difficult to determine the appropriate intensity level of a certain AU, meaning that they must go back to previous events to compare the relative intensity of an AU with other instances of that AU for a given person or multiple persons. Fast-FACS has the option (*mosaic*) to view all video events of a certain AU in one screenshot, with the capability to select the images with the same or similar intensity in order to code them all at the same time. By being able to compare multiple instances of an AU, users (coders) can directly calibrate intensity without having to hold multiple instances in mind. With the *event navigation* menu the coder can quickly verify that the event has been correctly coded, as well as change the estimated onset and offset if required. Fig. 3 (right) shows some of these functionalities.

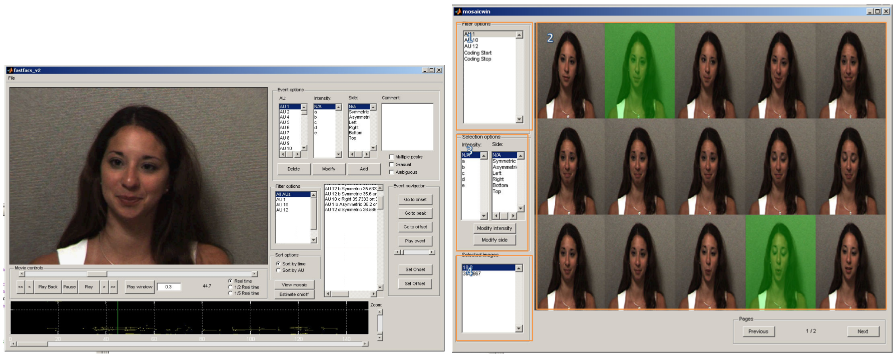


Fig. 3: Left) Fast-FACS main interface. Right) Details of mosaic window. 1.-AU list. 2.-Mosaic image. 3.-Intensity and side options. 4.-List of selected images.

4 Experimental evaluations

Fast-FACS enables computer-assisted coding of peaks and automatic coding of onsets and offsets. To evaluate Fast-FACS, at least three questions are relevant.

– How well does Fast-FACS compare with leading commercial software for manual coding of peaks? Inter-coder agreement should be comparable.
– Does automatic detection of onsets and offsets have concurrent validity with manual coding? Inter-system agreement for onsets and offsets should be comparable to inter-coder agreement of manual coding.
– Is Fast-FACS more efficient than manual coding? Does it substantially reduce the time required to complete FACS coding?

We conducted several experiments using a relatively challenging corpus of FACS coded video, the RU-FACS [14] video data-set. It consists of non-posed facial behavior of 100 participants who were observed for approximately 2.5 minutes. FACS-coded video from 34 participants was available to us. Of these, 5 had to be excluded due to excessive occlusion or errors in the digitized video, leaving data from 29 participants.

4.1 Inter-coder and inter-system agreement

Two sequences, S60 and S47, were selected at random from the RU-FACS database. The clips were 2 minutes 4 seconds and 2 minutes 50 seconds in duration, respectively. Each coder coded the two interviews using two software packages, Observer XT 7.0 [15], and Fast-FACS. AUs coded include AU 1, AU 2, AU 4, AU 10, AU 12, AU 15, and AU 20. Order of coding the clips was counter balanced across coders and across systems. Thus, for one subject, one coder used Fast-FACS first and Observer second, while the other coder began with Observer and then used Fast-FACS. The order was reversed for coding the other subject. Coding of the same clip was conducted several days apart to minimize the learning (familiarity) effect. The time it took each coder to code peaks in Observer and Fast-FACS was recorded. In addition, onset and offset of each AU were coded in Observer, and the time it took to code onset and offset was also recorded. Onsets and offsets in Fast-FACS were not manually coded, rather automatically estimated.

In calculating inter-coder and inter-system agreement, a window of agreement of $\pm.5$ seconds (15 frames) was used. In FACS coding, it is typical allowed a margin of error [16]. Inter-coder agreement [17] refers to whether two coders using the same system agree. Concurrent validity refers to whether there is agreement between systems. Percent agreement was computed using percentage agreement Ekman & Frisen [18], and as a coefficient Kappa (k) [19]. Kappa is a more rigorous metric as it is controls for agreements due to chance. Agreement is reported for both all intensity levels and for intensity levels B and higher. In the original FACS manual, AU at trace levels were not coded, and reliability of A levels has not been reported in the literature.

Intra and inter-system agreement for AU peaks: For both percent agreement and coefficient kappa when labeling peaks of all intensities, agreement between systems (86% and 74% kappa) was comparable to inter-coder agreement using commercial software (84% and 70% kappa) . When considering intensities B or higher, agreement rose to 83% and 91

Temporal agreement of manual coding of peak, onset and offset This section evaluated the inter-coder differences of manual coding. Temporal agreement of manual coding for peak, onset and offset was evaluated in Fast-FACS and Observer. The same two clips from the RU-FACS [14] database were used. The temporal error was calculated only when there was agreement between the two coders within a $\pm.5$ sec window. Results for temporal error using Fast-FACS and Observer are shown separately in Table 1a (left) and Table 1b (right). Both systems achieved similar results. On average, temporal error for manual coding of peaks and onset are about ± 2 frames. Temporal error for manual coding of offset was much larger, on average ± 10 frames in Observer and ± 12 frames in Fast-FACS. Across AU, average agreement was within $10 - 12$ frames. This finding is consistent with what is known from the FACS literature. In general, onsets are relatively discrete, whereas offsets for many AU fade gradually and may be difficult to delimit [5]. Also, it appears that temporal error of manual coding is different for different AUs, with AU 10 and 12 showing larger temporal error and greater variability than other AUs, especially for offset coding.

4.2 Accuracy of estimated onsets and offsets

To evaluate the accuracy of the onset and offset estimation we used 29 subjects from the RU-FACS database. Leave-one-out cross-validation was used in the evaluation, using all subjects except the one currently under test to train the detection algorithm (i.e. metric learning 3.3) and repeating the training/testing phases for every subject in the database. The detected onsets and offsets for all subjects were then pooled and compared with those coded manually, taken as ground truth.

As a baseline comparison and an intuition as to how much information the visual AAM features contribute to the estimation, Fig. 4 shows the performance of our system compared with a system that uses only the statistical distribution of AU onsets and offsets along time, and estimates the onset at the mean onset position (w.r.t. the peak) as found in the training data (similarly for offsets). Given that the temporal statistics of

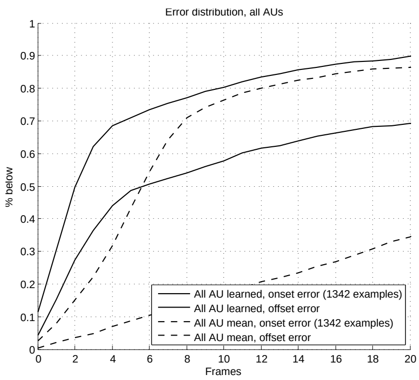
	Peak			Onset			Offset		
AU	M	SD	N	M	SD	N	M	SD	N
1	1.6	3.8	36	1.4	1.8	33	4.5	6.4	33
2	1.2	2.1	27	1.0	1.3	27	9.6	19.4	27
4	2.6	4.3	15	1.4	3.1	15	7.2	11.2	15
10	3.9	4.9	13	3.5	4.6	13	34.8	44.7	13
12	3.0	3.2	18	2.1	4.1	8	24.5	30.3	8
15	1.8	2.8	28	2.3	3.5	28	4.2	6.7	28
20	0.4	0.5	10	0.4	0.5	10	1.2	0.8	10
M	2.3			1.9			12.0		

(a) Fast-FACS

	Peak			Onset			Offset		
AU	M	SD	N	M	SD	N	M	SD	N
1	2.5	4.7	35	3.0	4.9	32	8.5	14.4	32
2	1.9	2.9	29	1.5	1.8	26	8.4	13.3	26
4	2.1	4.0	15	2.3	2.7	15	8.6	7.9	15
10	2.8	3.4	14	2.0	2.5	14	14.5	29.4	14
12	2.3	2.4	21	4.3	5.3	8	25.2	35.3	8
15	1.6	2.6	32	5.8	22.6	32	3.0	3.2	32
20	0.9	0.8	10	1.2	0.9	10	1.5	0.9	10
M	2.0			2.7			9.6		

(b) Observer

Table 1: Temporal error of manual peak, onset and offset in (a) Fast-FACS and (b) Observer. All units are in frames; M refers to mean, STD to standard deviation and N represents the number of samples.



(a) Error distribution

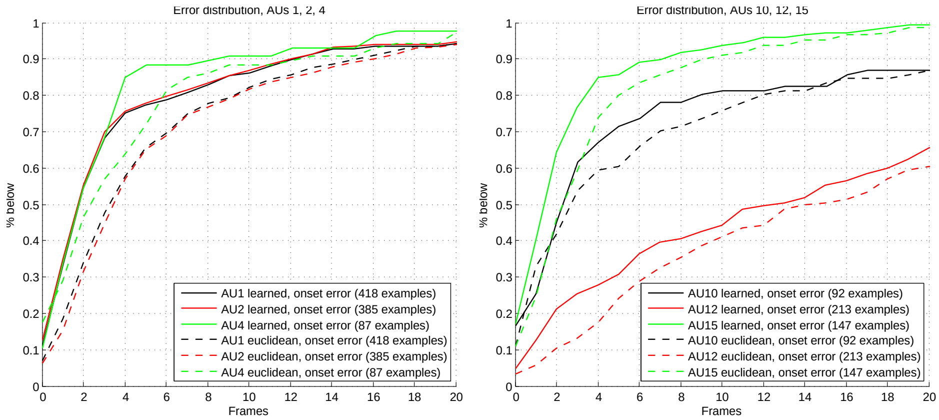
		S60		S47	
		Manual	Fast-FACS	Manual	Fast-FACS
C1	Peak	189	147	163	135
	Onset/Offset	137	0.0	113	0.0
C2	Peak	114	64	64	72
	Onset/Offset	66	0.0	62	0.0
Mean	Peak	151.5	105.5	113.5	103.5
	Onset/Offset	101.5	0.0	87.5	0.0
	% On/Off over	40%	0%	43%	0%
	total	253.0	105.5	200.0	103.5

(b) Manual coding time

Fig. 4: Left) Error distribution for onset and offset estimation using the learned metric compared with estimating the mean onset and offset duration (for each AU). Graph shows the fraction of instances in the database for which the error committed when estimating the onset and offset from the coded peak was below a certain number of frames. Results are pooled for all AUs (1,2,4,10,12,15,20). Right) Time required to code peaks, onsets, and offsets using manual coding via Observer and Fast-FACS. Onset and offset coding in Fast-FACS is fully automated, thus requiring no coder time. Units are in seconds. C1 refers to coder 1.

each AU can be different, these measures are taken for each AU separately. Note that for offsets especially, the visual features are key to an accurate estimation.

Figures 5a and 5b show the distribution of the errors committed for onsets of selected AUs, measured as the absolute difference in frames from the manually labeled onsets/offsets to those determined automatically, comparing the learned metric and Euclidean distance. Temporal error for onsets was relatively low. The mean error ranged



(a) AUs 1, 2, and 4

(b) AUs 10, 12, and 15

Fig. 5: Error distribution for onsets for (a) AUs 1, 2 and 4, (left) and (b) AUs 10, 12 and 15, comparing the learned metric with results obtained using Euclidean distance.

from 3.8 to 13.1 for AUs (1,2,4,10,15). This is within the standard of acceptable error for manual coding. Inter-system agreement for offsets was lower and more variable. With the exception of AU 12, mean offset error ranged from 10.6 to 36.2 frames. Mean onset error for AU 12 was 32.13 frames, mean offset error 84 frames. Lower precision for offsets is consistent with the reliability of manual coding. Many AU fade slowly as the face relaxes, which attenuates the observable signal values of offsets. Indeed, only in some selected types of smiles (e.g., embarrassed), does one find fast offsets [20]. Perhaps the most important confound for AU 12 is mouth movement due to speech, which makes similarity based methods fail.

4.3 Efficiency of Fast-FACS

Fast-FACS reduced total coding time by one half or more. As shown in the table in Fig. 4 right, automatic coding of onsets and offsets was responsible for most of this decrease. However, it also appeared that efficiencies in the Fast-FACS GUI may have been a factor as well. Manual coding of peaks using Fast-FACS was faster in each case. Overall, the increased efficiency from the GUI and from automatic coding of onsets and offsets resulted in dramatically increased efficiency and productivity.

5 Conclusion and future work

Fast-FACS has shown very good performance in experiments, and we believe that Fast-FACS shows great promise in providing a novel and easy tool for specific FACS event coding, and also providing a framework in which the development of computer-aided FACS tools such as automatic peak detection and verification is possible. Future work includes the incorporation of appearance features (key to extending the system to certain AUs, e.g. 14), as well as comparing other methods for onset and offset detection, particularly incorporating time to robustly account for changes due to speech.

References

1. P. Ekman and W. Friesen, "Facial action coding system: A technique for the measurement of facial movement." *Consulting Psychologists Press.*, 1978.
2. J. F. Cohn and T. Kanade, *Use of automated facial image analysis for measurement of emotion expression.* New York: Oxford.: The handbook of emotion elicitation and assessment. Oxford University Press Series in Affective Science., 2007.
3. M. Bartlett, G. Littlewort, C. Lainscsek, I. Fasel, M. Frank, and J. Movellan, "Fully automatic facial action recognition in spontaneous behavior." in *7th International Conference on Automatic Face and Gesture Recognition.*, 2006.
4. M. Pantic and I. Patras, "Dynamics of Facial Expression: Recognition of Facial Actions and their Temporal Segments from Face Profile Image Sequences," *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol. 36, no. 2, pp. 433–449, April 2006.
5. J. Cohn and P. Ekman, "Measuring facial action by manual coding, facial emg, and automatic facial image analysis." *J. A. Harrigan, R. Rosenthal and K. Scherer (Eds.). Handbook of nonverbal behavior research methods in the affective sciences. NY: Oxford.*, 2005.
6. M. Valstar and M. Pantic, "Combined support vector machines and hidden markov models for modeling facial action temporal dynamics," in *Proceedings of IEEE Workshop on Human Computer Interaction. In conjunction with IEEE ICCV-07.*, 2007, pp. 118–127.
7. Y. T. L. Zhang and Q. Ji, "Active image labeling and its application to facial action labeling," in *European Conference on Computer Vision*, 2008.
8. F. de la Torre, J. Campoy, Z. Ambadar, and J. Cohn, "Temporal segmentation of facial behavior," in *International Conference on Computer Vision*, 2007.
9. F. De la Torre and M. Nguyen, "Parameterized kernel principal component analysis: Theory and applications to supervised and unsupervised image alignment," in *CVPR*, 2008.
10. T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," in *European Conference Computer Vision*, 1998, pp. 484–498.
11. I. Matthews and S. Baker, "Active appearance models revisited," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 135–164, Nov. 2004.
12. E. Xing, A. Ng, M. Jordan, and S. Russell, "Distance metric learning, with application to clustering with side-information," in *Advances in Neural Information Processing Systems 15*. MIT Press, 2003, pp. 505–512.
13. M. Grant and S. Boyd, "Cvx: Matlab software for disciplined convex programming," http://stanford.edu/~boyd/cvx, 2008.
14. M. Frank, J. Movellan, M. Bartlett, and G. Littleworth, "Ru-facs-1 database,machine perception laboratory." *U.C. San Diego*.
15. "http://www.noldus.com/."
16. M. A. Sayette, J. F. Cohn, J. M. Wertz, M. Perrott, and D. J. Parrott, "A psychometric evaluation of the facial action coding system for assessing spontaneous expression." *Journal of Nonverbal Behavior*, no. 25, pp. 167–186, 2007.
17. J. Coan and J. Allen, *Oxford handbook on emotion elicitation and assessment.* NY: Oxford., (In Press).
18. P. Ekman, W. Friesen, and J. Hager, "Facial action coding system: Research nexus." *Network Research Information*, 2002.
19. J. Fleiss, *Statistical methods for rates and proportions*. Wiley, New York, 1981.
20. Z. Ambadar, J. Cohn, and L. Reed, "All smiles are not created equal: Morphology and timing of smiles perceived as amused, polite, and embarrassed/nervous," *Journal of Nonverbal Behavior*, 2008.