Efficient Measuring of Facial Action Unit Activation Intensities using Active Appearance Models

Daniel Haase[1], Michael Kemmler[1], Orlando Guntinas-Lichius[2], Joachim Denzler[1]

[1] Computer Vision Group, Friedrich Schiller University of Jena, Germany
[2] Department of Otolaryngology, University Hospital Jena, Germany

Abstract

The Facial Action Coding System describes a set of 44 ordinally scaled actions units (AUs), which are used to create facial expressions. In medical applications such as the therapy of a facial paralysis, automatically finding the activation intensity of each AU is of main interest. In this medical application context, existing works feature several drawbacks. For instance, the majority of approaches concentrates on analyzing the presence or absence of AUs instead of finding their underlying activation levels. For facial feature extraction, Active Appearance Models (AAMs) are often used. While all current AAM-based studies rely on derived redundant high-dimensional features, we propose to use a compact low-dimensional representation. Exhaustive experiments using three widely-used datasets and several prediction methods demonstrate the benefits of our approach. While retaining a state-of-the-art recognition accuracy, our approach is substantially faster and thus perfectly suited for analyzing large-scale databases.

1 Introduction and Related Work

The identification of facial expressions is a key element in several problems from psychology and cognitive science [19]. Each facial expression is the result of a unique interaction of 43 facial muscles. For medical applications, *e.g.* when dealing with pathological dysfunctions such as a unilateral partial and complete facial paralysis, identifying these *individual* activations rather than the corresponding expression is of interest. Applications for this scenario range from aiding a diagnosis to supporting a therapy. A prominent way of describing the underlying activations is the *Facial Action Coding System* (FACS) introduced by Ekman and Friesen [6, 7]. It describes facial actions based on a set of 44 fundamental *Action Units* (AUs) such as "raise of inner eyebrows" (AU 1) or "depressing of lip corners" (AU 15). The degree of activation or *intensity* of each AU is measured on an ordinal scale ranging from A (barely observable) to E (physical maximum). However, the manual FACS annotation of facial images is very time-consuming and requires highly trained human experts. For this reason, a fully automated FACS classification system is extremely desirable.

For automated facial expression and AU analysis, an immense amount of literature exists in the field of computer vision, *e.g.* [8, 21, 12, 9]. It can mainly be distinguished between local and global methods of feature extraction [8]. One example for the former is [20], where the area around the eyes and the corresponding AUs are analyzed based on local binary patterns. However, if the entire set of AUs is to be predicted, global approaches are more promising [8]. *Active Appearance Models* (AAMs) [4] are particularly suited for this task, as they generate compact parameterized models which

(a) MPAFC dataset   **(b)** CK+ dataset
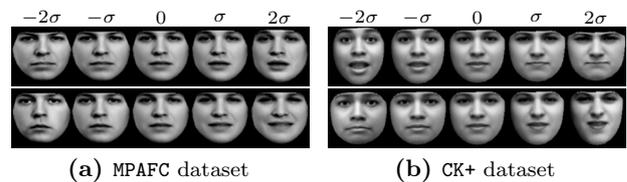
Figure 1. Examples for the influence of combined AAM parameters. For two parameters (top/bottom row), the resulting model instances are shown for AAMs trained on two different datasets (MPAFC and CK+).

describe the appearance of object instances shown in images, *e.g.* faces. An example of the influence of AAM parameters can be seen in Figure 1, where synthesized model instances for varying parameter values are shown. AAMs are successfully applied for both facial expression recognition [11], as well as AU detection [13, 22].

However, in the context of a medical application as described above, all current approaches to AU prediction suffer from at least one of the following shortcomings:

- Instead of considering the entire AU intensity scale, *only AU presence/absence is predicted* [13, 20, 22]. To assess, compare or document the degree of a facial paralysis, this is highly insufficient.
- Very *high-dimensional, redundant features* are used [14, 2, 13]. For large-scale medical studies with several thousand images, this leads to unreasonably large computation times.
- Prediction is *solely performed for groups of AUs* [20], and thus is unsuitable to observe individual muscle activations.
- Only *small subsets of AUs are considered* [18, 15, 20] which leads to a limited application range.

This paper adresses all those issues. Our approach is based on a low-dimensional representation directly derived from AAMs that enables a fast and accurate prediction of AU intensities. Exhaustive experiments on three well-established AU datasets are conducted, validating the suitability of our approach. We show that, while retaining a state-of-the-art recognition accuracy, our approach is substantially faster and thus perfectly suited for analyzing large-scale databases.

The remainder of this paper is structured as follows. In Sect. 2, a short introduction to AAMs is given. Prediction methods which are used in this work, are presented in Sect. 3. A detailed description about our AU intensity prediction scheme is given in Sect. 4 along with experiments validating its suitability. Finally, Sect. 5 summarizes and concludes the paper.

2 Active Appearance Models

This section gives a brief overview of *Active Appearance Models* (AAMs) [4], which are used for FACS prediction in this paper.

2.1 Model Training

Given N annotated images $\boldsymbol{I}_1, \ldots, \boldsymbol{I}_N$ showing instances of an object class such as faces, AAMs can be learnt automatically. The result is a parameterized appearance model, which can be fitted to new images.

Firstly, the vectorized shapes $\boldsymbol{l}_1, \ldots, \boldsymbol{l}_N$ are aligned according to translation, rotation, and scale and combined to the matrix $\boldsymbol{L} = (\boldsymbol{l}_1 - \boldsymbol{l}_\mu, \ldots, \boldsymbol{l}_N - \boldsymbol{l}_\mu)$, where $\boldsymbol{l}_\mu = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{l}_n$ is the *mean shape*. Applying Principle Component Analysis (PCA) to $\boldsymbol{L}$ yields the matrix $\boldsymbol{P_L}$ of *shape eigenvectors*, and each shape $\boldsymbol{l}'$ can be expressed by its *shape parameters* $\boldsymbol{b}_{l'}$ via

$$\boldsymbol{l}' = \boldsymbol{l}_\mu + \boldsymbol{P_L} \boldsymbol{b}_{l'}, \quad \text{where} \quad \boldsymbol{b}_{l'} = \boldsymbol{P_L}^\top \left(\boldsymbol{l}' - \boldsymbol{l}_\mu \right). \quad (1)$$

In the second step, the object textures of each training image are shape-normalized to form the texture vectors $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_N$. As for the shape model, PCA is applied to obtain the *texture eigenvectors* $\boldsymbol{P_G}$, and the *texture parameters* $\boldsymbol{b}_{g'}$ are defined similarly to Eq. 1.

As last step, shape and texture submodels are combined to form the appearance model. Correlations between shape and texture, *i.e.* the *appearance eigenvectors* $\boldsymbol{P_C}$, are revealed by applying PCA on the set of the variance-weighted (factor w) and concatenated shape and texture parameters. Using $\boldsymbol{P_C}$, the shape and texture of an object instance having the parameters $\boldsymbol{c}' = [w\boldsymbol{b}_{l'}^\top, \boldsymbol{b}_{g'}^\top]^\top$ can be expressed by

$$\boldsymbol{c}' = \boldsymbol{P_C} \boldsymbol{b}_{c'}, \quad \text{where} \quad \boldsymbol{b}_{c'} = \boldsymbol{P_C}^\top \boldsymbol{c}' \quad (2)$$

are the *appearance parameters*. The complexity of the model can be vastly reduced by only using eigenvectors which explain a minimum amount of variance. In Figure 1, the influence of appearance parameters is exemplarily shown for two AAMs trained on face images.

2.2 Model Fitting

To fit a trained AAM to new images, the appearance parameters are adapted to minimize the texture difference $\Delta \boldsymbol{g}$ between model and image. The relationship between model parameter updates $\Delta \boldsymbol{b}$ and $\Delta \boldsymbol{g}$ is modeled via $\Delta \boldsymbol{b} = \boldsymbol{R} \Delta \boldsymbol{g}$. Here, $\boldsymbol{R}$ is the matrix of linear coefficients which can be estimated by systematically displacing the known parameters of the training images.

3 Prediction Methods

This section briefly reviews two non-standard prediction methods used in our work. More details about all considered prediction methods and their role for inferring AU intensities are provided in Sect. 4. Given a training set $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{Y})$ with inputs $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^\top \in \mathcal{X}^N$ and outputs $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N]^\top \in \mathcal{Y}^N$, the goal is to realize a mapping from input to output space that generalizes to unseen data $\boldsymbol{x}_* \in \mathcal{X}$.

3.1 Gaussian Process Regression

In *Gaussian process (GP) regression*, univariate outputs $y \in \mathbb{R}$ are assumed to be generated according to $y(\boldsymbol{x}) = f(\boldsymbol{x}) + \varepsilon$, where f is a latent function and ε denotes a noise component. Bayesian methods for solving above problem aim at specifying priors for f and ε. Based on the provided uncertainties, inference is possible by integrating out these non-observed variables. In GP regression, the prior probability is a Gaussian process using mean function $\mu : \mathcal{X} \to \mathbb{R}$ and covariance (kernel) function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, *i.e.* $f \sim \mathcal{GP}(\mu, \kappa)$,

Table 1. Overview of the FACS datasets used in this paper. Only samples with *intensity* rated FACS data were used. For the leave-one-out tests, only AUs with more than one activated sample were considered.

Dataset		Ek60 [5, 23]	MPAFC [19]	CK+ [10, 13]
Intensity FACS Ratings	Subjects	10	8	73
	Images	60	128	116
	Covered AUs	31	40	24
	AUs with ≥2 Samples	25	32	22
AAM	Landmarks	58	58	68
	Texture Size	7,564	5,215	7,013
	Explained Variance	95.16%	95.14%	95.02%
	Comb. Parameters	30	46	40

which can be seen as a normal distribution over functions. By assuming *i.i.d.* zero-mean Gaussian noise $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$, outputs y_* follow a normal distribution [17] with moments $\overline{y}_* = \boldsymbol{y}^\top \left(\boldsymbol{K} + \sigma_n^2 \boldsymbol{I} \right)^{-1} \boldsymbol{k}_*$ and $\sigma_*^2 = k_{**} - \boldsymbol{k}_*^\top \left(\boldsymbol{K} + \sigma_n^2 \boldsymbol{I} \right)^{-1} \boldsymbol{k}_* + \sigma_n^2$, where shortcuts $\boldsymbol{K} = \kappa(\boldsymbol{X}, \boldsymbol{X})$, $\boldsymbol{k}_* = \kappa(\boldsymbol{X}, \boldsymbol{x}_*)$, and $k_{**} = \kappa(\boldsymbol{x}_*, \boldsymbol{x}_*)$ were used. Predictions can be made based on estimate $\overline{y}_*$. For multiple output prediction, sharing the kernel is computationally attractive, yielding estimates $\overline{\boldsymbol{y}}_* = \boldsymbol{Y}^\top \left(\boldsymbol{K} + \sigma_n^2 \boldsymbol{I} \right)^{-1} \boldsymbol{k}_*$. Hyperparameters of κ can be learned by marginal likelihood maximization.

3.2 Cumulative Link Models

Cumulative link models [1] aim to infer outputs $y \in \mathcal{Y} = [1, \ldots, C]$ which obey $1 \leq \ldots \leq C$. In order to estimate probabilities $\pi_{ik} = P(y_k = i)$, assumptions are placed upon their cumulative probabilities, *e.g.* $P(y_k \leq i) = \sum_{j \leq i} \pi_{jk} = g^{-1}(\boldsymbol{b}^\top \boldsymbol{x}_k + a_i)$. The *canonical link function* $g : [0, 1] \to \mathbb{R}$ is usually a sigmoid, such as $g(z) = [1 + \exp(-z)]^{-1}$. Parameters are estimated by maximum likelihood estimation. Multi-variate outputs $\boldsymbol{Y}$ can be estimated using m univariate models.

4 Experiments and Results

In the following, details about used FACS databases and insights into our prediction and evaluation scheme are provided. Finally, a comparison to the state-of-the-art in terms of accuracy and runtime behavior is done, showing the suitability of our approach.

4.1 Datasets

Three datasets were used to investigate the FACS prediction performance of the presented methods. In particular, these datasets were the *Ekman and Friesen Series* (Ek60) [5, 23], the *Montréal Pain and Affective Face Clips* (MPAFC) [19] and the *Extended Cohn-Kanade Dataset* (CK+) [10, 13]. For all datasets, only those samples containing *intensity*, *i.e.* five point ordinal scale FACS ratings were selected, resulting in sample sizes of 60, 128, and 116 images. The Ek60 dataset consists of images of ten persons performing six basic emotions anger, sadness, surprise, disgust, fear, and happiness. In the MPAFC dataset, these six emotions plus pain and neutral are performed by eight persons in a total of 64 video clips. For each video clip, the first (ideally neutral expression) and last (target expression) frame is FACS coded, resulting in 128 images. The well-known CK+ dataset contains 593 sequences, of which a subset of 116 samples does have intensity FACS ratings.

For AAM training, landmark data was manually created for the Ek60 and MPAFC datasets, whereas for CK+ available ground truth landmarks were used. An overview of the used datasets can be found in Table 1.

4.2 AU Prediction and Evaluation Framework

The task of AU intensity prediction can be cast as a collection of independent prediction problems, each concentrating on a single AU. We rely on combined AAM parameters $\boldsymbol{b}_{c'}$ from Equation (2), which stands in contrast to other existing approaches for AU intensity estimation [14, 2, 13]. These combined texture and shape features are then used as inputs for the prediction methods outlined in Sect. 3 to learn the relationship to available AU intensities. The latter are given by the five point FACS codings and identified with numbers 0 (not present) as well as 1, . . . , 5 (intensities A, . . . , E).

For evaluation, leave-one-out (LOO) cross-validation is carried out separately for each dataset. The particular AAM which is used for each dataset is obtained by training on all images of the respective dataset. Then, each AU is analyzed separately, based on the *confusion matrix* obtained by the LOO test. Note that, as stated in Table 1, some AUs are only activated in exactly one sample. To avoid biases in the results, such AUs are not considered in the LOO tests. By summing over all AUs, a *total confusion matrix* is retrieved.

Based on these confusion matrices, various types of recognition rates can be derived. The most obvious error measure is to only count a prediction as correct if a predicted AU activation is identical to its ground truth counterpart (measure "exact"). Taking the ordinal nature of the FACS ratings into account, a suitable measure of divergence can be obtained by allowing differences of one intensity level between prediction and ground truth ("diff1"). Another interesting aspect is whether active and inactive AUs are predicted as such, disregarding the different intensities of active AUs ("onOff"). The number of samples for each AU intensity varies drastically, hence we base our evaluations on the *average* recognition rate over all AU intensities. In addition to GP regression (gpr) and cumulative link models (clm), we also analyzed multivariate linear and non-linear regression (mlr, nls), linear support vector machines (lsvm) and nearest neighbor (1nn).

Implementation Details. All prediction steps were carried out in the programming language R. Own implementations for AAMs and GP regression were used with settings listed in Table 1 and kernel $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \theta_1^2 \exp(-||\boldsymbol{x} - \boldsymbol{x}'||^2/(2\theta_2^2))$. For (non-)linear regression, standard R functions were used. Non-linearity was achieved via $\Phi(\boldsymbol{x}) = [\boldsymbol{x}^\top, \tilde{\boldsymbol{x}}^\top]^\top$, where $\tilde{\boldsymbol{x}}$ contains squared input values. For clm and lsvm, packages ordinal and kernlab were utilized. Trade-off parameters $C = 10^{-5}, 10^{-4}, \ldots, 10^5$ were considered for lsvm.

4.3 Results

The total average recognition rates obtained by LOO cross-validation for all datasets are shown in Table 2 (top). For each dataset and method, the three numbers are obtained by using the error measures explained above. It can be stated that the order of magnitude for corresponding numbers is the same for all three datasets. In addition, the ranking of the methods for each error measure is roughly identical. Across all datasets, mlr and gpr always give the best results except for one case. Note that gpr is the best method for the CK+ dataset, which by far has the most number of subjects (see Table 1). This suggests to use GP regression in the general prediction case. The next best method is clm, which performs a true ordinal regression and therefore is to be expected to give good results espe-

©Jeffrey Cohn

| 1C 2C 4B 5B 12A | 20C 25B | | 1C 2C 5D | 25D 26D |

| 1A 2A 4B 5A 12B 14A 20A 25B 26B | | **True Prediction** | 1B 2B 5B 16A 23A 25C 26C 27A 38A |
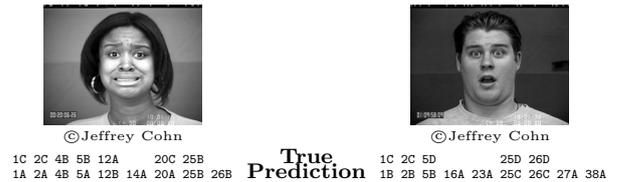
Figure 2. Prediction examples for images from the CK+ dataset (subjects S125/006 and S132/008). For prediction, method gpr was used. The model was trained on all images of the dataset except for the ones depicted above. AU activations are coded in the standard notation.

cially for the error measures exact and diff1. Method nls is inferior to mlr, presumably because non-linear regression has a larger set of parameters which may lead to overfitting and less robust parameter estimation. Despite the different classifier capacity, 1nn and lsvm give comparable results. An example prediction using gpr on the CK+ dataset is shown in Figure 2.

Cross-database tests were carried out by using different datasets for AAM training and FACS prediction. The resulting recognition rates are about five to ten percentage points lower compared to those listed in Table 2. For the case that *ideal* rather than fitted AAM parameters are used for prediction, all recognition rates slightly increase by about two to five percentage points.

Comparison to the State-of-the-art in AU Intensity Estimation. The previous experiment validates that encouraging results can be obtained using our proposed features. To better quantify their suitability, we additionally compared our approach to the work of [14, 2, 13]. Instead of using low-dimensional appearance parameters, the authors rely on a concatenation of both landmarks (SPTS) and transformed pixel intensities (CAPP) in order to capture shape and texture information. Applied to our experimental setup, this gives rise to inputs of dimensionality 7,680, 5,331, and 7,149 for datasets Ek60, MPAFC, and CK+, respectively.

The resulting recognition rates are displayed in Figure 2 (bottom). Instead of solely using linear SVMs as done in [14, 2, 13], we also employed mlr, gpr, and 1nn. The remaining methods were omitted due to the associated computational demands. It becomes apparent that mlr and gpr again robustly achieve accurate recognition results, outperforming lsvm in nearly all cases. The worst performance is given by 1nn. There is no improvement in overall performance by employing the high-dimensional representations of [14, 2, 13], with the notable exception of using lsvm. All other methods tend to produce comparable or even inferior results. This indicates that our approach of using AAM parameter features is to be preferred in terms of accuracy.

Runtime Analysis. Average training and prediction runtimes of our framework using both our low-dimensional features and state-of-the-art high-dimensional features (SPTS+CAPP) [14, 2, 13] are provided in Table 2d. Note that prediction times are given for infering the entire set of AUs for one input image, measured on a desktop computer with an Intel® Core™ i5 760 CPU (2.80 GHz). The key result is that runtimes for training and prediction are substantially reduced by employing our low-dimensional representation. Moreover, certain methods such as nls and clm could not be considered for high-dimensional features due to exceeding computational demands. Using "project out" AAM fitting which requires less than 5 ms per image [16] and our GP implementation with hyper-

Table 2. Total average recognition rates for leave-one-out cross-validation on the three FACS datasets Ek60, MPAFC and CK+. The prediction methods used were *cumulative link models* (clm), *Gaussian process regression* (gpr), *linear regression* (mlr), *non-linear regression* (nls), *nearest neighbor* (1nn) and *linear SVMs* (lsvm).

		(a) Ek60			(b) MPAFC			(c) CK+			(d) Comput. Times	
	Prediction Method	Error Measure			Error Measure			Error Measure			Training (per Dataset)	Prediction (per Image)
		exact	diff1	onOff	exact	diff1	onOff	exact	diff1	onOff		
Our Approach	mlr	**42.2%**	**76.0%**	**85.7%**	35.5%	**67.6%**	**79.4%**	31.8%	64.2%	85.1%	0.01 s	8.67 ms
	gpr	40.1%	75.2%	84.8%	33.7%	67.0%	79.1%	**32.3%**	**66.6%**	**86.3%**	1.17 s	7.45 ms
	clm	29.4%	62.9%	72.7%	**37.3%**	66.5%	66.1%	30.1%	57.7%	66.4%	1.39 s	182.96 ms
	nls	20.1%	48.3%	57.4%	31.0%	59.0%	71.3%	29.4%	58.9%	75.8%	1.17 s	245.16 ms
	lsvm	30.1%	62.3%	62.1%	28.2%	52.8%	58.4%	27.8%	57.8%	64.3%	12.09 s	22.82 ms
	1nn	30.1%	62.4%	65.5%	27.2%	48.2%	42.1%	29.6%	58.5%	61.0%	0.01 s	8.61 ms
SPTS+ CAPP [2, 13]	mlr	41.6%	**76.3%**	**83.6%**	35.2%	**66.8%**	**79.7%**	**31.6%**	64.2%	**85.8%**	385.95 s	31.60 ms
	gpr	**42.8%**	74.0%	83.5%	32.9%	66.3%	79.0%	30.3%	**64.4%**	85.0%	38.78 s	60.97 ms
	lsvm	33.7%	66.6%	68.7%	**35.9%**	61.8%	63.4%	31.1%	60.4%	64.9%	40.79 s	2082.21 ms
	1nn	26.8%	52.9%	56.2%	24.4%	46.4%	38.0%	27.0%	60.5%	58.3%	0.20 s	63.94 ms

parameter sharing, the total runtime can be reduced to below 13 ms per image. Compared to existing AAM and lsvm-based approaches such as [14, 2, 13] this is a tremendous speed-up allowing for large-scale medical applications and studies.

5 Conclusions and Further Work

This work concentrated on measuring facial action unit (AU) activation intensities, which are required in several medical scenarios. Based on the framework of Active Appearance Models, we proposed a simple but powerful representation that enables efficient large-scale analyses of face images. Our approach eliminates shortcomings of existing approaches, which use highly redundant features or solely predict the presence and absence of AU activations, thereby ignoring the underlying ordinal scale. Extensive experiments were conducted on three prominent datasets, each comprising at least 22 AUs. A comparison to state-of-the-art approaches clearly revealed the benefits of our approach: while retaining good accuracy, it is substantially faster, especially when using sophisticated prediction methods such as Gaussian process (GP) regression.

Given the success of GP regression and the nature of AU intensities, it would be interesting to incorporate *ordinal* GP regression [3] into future analyses. We further believe that improvements can be achieved by including a dynamical aspect into AU intensity prediction.

References

[1] A. Agresti. *Analysis of Ordinal Categorial Data.* Wiley, 2nd edition, 2010.

[2] A. B. Ashraf, S. Lucey, J. F. Cohn, T. Chen, Z. Ambadar, K. M. Prkachin, and P. E. Solomon. The painful face - pain expression recognition using active appearance models. *Im. Vis. Comp.*, 27(12):1788–1796, 2009.

[3] W. Chu and Z. Ghahramani. Gaussian processes for ordinal regression. *Mach. Learn. Res.*, 6:1019–1041, 2005.

[4] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *Patt. Anal.*, 23(6):681–685, 2001.

[5] P. Ekman and W. Friesen. *Pictures of facial affect.* Consulting Psych. Press, 1976.

[6] P. Ekman and W. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement.* Consulting Psych. Press, 1978.

[7] P. Ekman, W. Friesen, and J. Hager. *Facial Action Coding System (FACS).* A Human Face, 2002.

[8] B. Fasel and J. Luettin. Automatic facial expression analysis: a survey. *Patt. Recogn.*, 36(1):259–275, 2003.

[9] B. Jiang, M. F. Valstar, and M. Pantic. Action unit detection using sparse appearance descriptors in space-time video volumes. In *Autom. Face and Gesture Recogn.*, pages 314–321, 2011.

[10] T. Kanade, Y. Tian, and J. F. Cohn. Comprehensive database for facial expression analysis. In *Autom. Face and Gesture Recogn.*, pages 46–53, 2000.

[11] A. Lanitis, C. J. Taylor, and T. F. Cootes. Automatic interpretation and coding of face images using flexible models. *T. Pattern Anal.*, 19:743–756, 1997.

[12] G. Littlewort, J. Whitehill, T. Wu, I. R. Fasel, M. G. Frank, J. R. Movellan, and M. S. Bartlett. The computer expression recognition toolbox (CERT). In *Autom. Face and Gesture Recogn.*, pages 298–305, 2011.

[13] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops*, pages 94–101, 2010.

[14] S. Lucey, A. Ashraf, and J. Cohn. *Investigating spontaneous facial action recognition through AAM representations of the face.* 2007.

[15] M. H. Mahoor, S. Cadavid, D. S. Messinger, and J. F. Cohn. A framework for automated measurement of the intensity of non-posed facial action units. In *Computer Vision and Pattern Recognition Workshops*, 2009.

[16] Iain Matthews and Simon Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, 2004.

[17] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning.* The MIT Press, 2005.

[18] J. Reilly, J. Ghent, and J. McDonald. Investigating the dynamics of facial expression. In *Symposium on Advances in Visual Computing*, pages 334–343, 2006.

[19] D. Simon, K. D. Craig, F. Gosselin, P. Belin, and P. Rainville. Recognition and discrimination of prototypical dynamic expressions of pain and emotions. *Pain*, 135(1–2):55–64, 2008.

[20] R. S. Smith and T. Windeatt. Facial expression detection using filtered local binary pattern features with ECOC classifiers and platt scaling. *J. Mach. Learn. Res.*, 11:111–118, 2010.

[21] Y. Tong, W. Liao, and Q. Ji. Facial action unit recognition by exploiting their dynamic and semantic relationships. *IEEE TPAMI*, 29(10):1683–1699, 2007.

[22] L. van der Maaten and E. Hendriks. Action unit classification using active appearance models and conditional random fields. *Cogn. Process.*, 13(Suppl 2):S507–S518.

[23] A. W. Young, D. I. Perrett, A. J. Calder, and P. Sprengelmeyer, R.and Ekman. *Facial expressions of emotion: stimuli and tests.* Thames Valley Test Co., 2002.