

AAM Derived Face Representations for Robust Facial Action Recognition

Simon Lucey, Iain Matthews, Changbo Hu, Zara Ambadar,
Fernando de la Torre, Jeffrey Cohn
Robotics Institute, Carnegie Mellon University
Pittsburgh PA 15213, USA

slucey@ieee.org, iainm@cs.cmu.edu, changbo@cs.cmu.edu,
ambadar@pitt.edu, ftorre@cs.cmu.edu, jeffcohn@cs.cmu.edu

Abstract

In this paper, we present results on experiments employing Active Appearance Model (AAM) derived facial representations, for the task of facial action recognition. Experimental results demonstrate the benefit of AAM-derived representations on a spontaneous AU database containing “real-world” variation. Additionally, we explore a number of normalization methods for these representations which increase facial action recognition performance.

1. Introduction

The Facial Action Coding System (FACS) [1–3] is the leading method for measuring facial movement in behavioral science. FACS has been successfully applied, but not limited to, identifying the differences between simulated and genuine pain, differences between when people are telling the truth versus lying, and differences between suicidal and non-suicidal patients. Typically, FACS is performed manually by trained human experts.

With the advent of modern computer vision and pattern recognition techniques, considerable progress¹ has been made towards the lofty goal of “real-world” automatic facial action recognition. However, for any pattern recognition task there always exists a constant conflict between within-class and between-class variation. The task of the scientist/engineer is to design a facial action recognition system that maximizes between-class variation while minimizing within-class variation.

We investigated the relative performance of a broad gamut of facial representations on a database that is representative of spontaneous facial behavior. These multiple facial representations are derived from a model-based approach for modeling and tracking the face. This approach is referred to in literature as an Active Appearance Model (AAM) [5]. Results demonstrate that AAM derived representations to be of real benefit for spontaneous facial behav-

ior when a number of “real-world” within-class variations occur. Additionally, we explore a number of normalization methods for these representations which increase facial action recognition performance.

2. Background

One of the first studies into representations of the face, for automatic facial action recognition, was conducted by Donato et al. [3]. Motivated by the plethora of work previously performed in the face recognition community, this study was restricted to only 2-D appearance based representations of the face (e.g. raw pixels, optical flow, Gabor filters, etc.) as well as data-driven approaches for obtaining compact features (e.g. PCA, LDA, ICA, etc.). These appearance based approaches were broadly categorized into monolithic and parts based representations. That is, where patches of pixels within the face are either analyzed holistically (monolithic) or locally (parts). In the ensuing literature, appearance based approaches have continued to be popular as demonstrated by the recent feature evaluation paper by Bartlett et al. [6]. A major criticism of purely appearance based approaches however, is their lack of shape registration. When “real-world” variation tends to occur, their lack of shape registration (i.e. knowledge of the position of the eyes, brow, mouth, etc.) can make normalizing for this variation very difficult.

Model-based approaches offer an alternative to appearance based approaches for representing the face. Typical approaches have been Active Shape Models (ASMs) [7] and Active Appearance Models (AAMs) [7] where both appearance and shape can be extracted and decoupled from one another. Model-based approaches, like those seen in an AAM, have an inherent benefit over purely appearance based approaches in the sense they can account and attempt to normalize many types of “real-world” variation. They are however, limited in some circumstances by their ability to accurately register the face in terms of shape and deal with 3D variation.

¹Readers are encouraged to inspect the following review articles [3, 4].

3. Database and Scope

3.1. Database

Our experiments were carried out on video data-set, recorded at Rutgers University², containing spontaneous AUs where 33 people of varying ethnicity and sex lied or told the truth to an interviewer. Data from 20 of the subjects was available for our experiments, containing 4 males and 16 females. Of the 20 subjects, 13 were of European heritage, 1 of African heritage, and 6 Asian. There was some degree of head movement in the data-set with frontal and out-of-plane head motion being common.

Lies were elicited in the following manner. A situation was created where the subject entered a room where there was a check, for a specified amount (typically \$100) made to an organization that the subject opposes. The subject was then interviewed about whether he/she took the check. The subject has to convince the interviewer that he/she did not take the check, even if they did. If the subject is believed, the check will be donated to an organization that he/she supports. If not, the check is donated to an organization he/she vehemently disagrees with. Interviews typically lasted 5-7 minutes and consisted of around 13 questions.

3.2. Scope

The scope of this paper was restricted to the specific task of peak-to-peak AU recognition. Typically, when an AU is annotated there is an onset, offset and peak time stamp associated with an individual AU sequence. Certified FACS coders at the video collection site manually FACS-coded the whole interview section. A certified FACS coder from the University of Pittsburgh verified the codes. Sequences for which manual FACS coding was not confirmed were excluded. Only AUs of at least intensity B were employed in our experiments. Onset and offset time stamps were assumed to be representative of a local AU 0 (i.e. neutral expression). AU 0 is employed later in our experiments in a normalization technique.

Contraction of the facial muscles produces changes in the appearance and shape of facial landmarks (e.g., brows) and in the direction and magnitude of motion on the surface of the skin and in the appearance of transient facial features (e.g., wrinkles). In our experiments, we focus on two types of muscle action. Contraction of the frontalis muscle raises the brows in an arch-like shape (AU 1 in FACS) and produces horizontal furrows in the forehead (AU 1+2 in FACS). Contraction of the corrugator supercilii and depressor supercilii muscles draws the inner (i.e., medial) portion of the brows together and downward and causes vertical wrinkles to form or deepen between the brows (AU 4 in FACS). The levator palpebrae superioris (AU 5 in FACS) is associated with the raising of the upper eyelid. Because these action units and action unit combinations in the eye and brow re-

gion occur frequently during conversation and in expression of emotion, we concentrated on them in our experiments.

Our experiments were conducted on the task of *subject-independent* facial action recognition. Due to the meagre number of examples available, a leave-one-out approach [8] had to be employed during our experiments.

4. AAM Derived Representations

In this section we describe both 2D and 3D active appearance models. We later derive features based on both models. Please note examples of the AAM shape registration results in Figure 1.

4.1. 2D Active Appearance Models

The *2D shape* \mathbf{s} of a 2D AAM [7] is a 2D triangulated mesh. In particular, \mathbf{s} is a column vector containing the vertex locations of the mesh. AAMs allow linear shape variation. This means that the 2D shape \mathbf{s} can be expressed as a base shape \mathbf{s}_0 plus a linear combination of m shape vectors \mathbf{s}_i :

$$\mathbf{s} = \mathbf{s}_0 + \sum_{i=1}^m p_i \mathbf{s}_i \quad (1)$$

where the coefficients $\mathbf{p} = (p_1, \dots, p_m)^T$ are the shape parameters. AAMs are normally computed from training data consisting of a set of images with the shape mesh (hand) marked on them [7]. The Procrustes alignment algorithm and Principal Component Analysis (PCA) are then applied to compute the base shape \mathbf{s}_0 and the shape variation \mathbf{s}_i .

The *appearance* of a 2D AAM is defined within the base mesh \mathbf{s}_0 . Let \mathbf{s}_0 also denote the set of pixels $\mathbf{u} = (u, v)^T$ that lie inside the base mesh \mathbf{s}_0 , a convenient abuse of terminology. The appearance of the AAM is then an image $A(\mathbf{u})$ defined over the pixels $\mathbf{u} \in \mathbf{s}_0$. AAMs allow linear appearance variation. This means that the appearance $A(\mathbf{u})$ can be expressed as a base appearance $A_0(\mathbf{u})$ plus a linear combination of l appearance images $A_i(\mathbf{u})$:

$$A(\mathbf{u}) = A_0(\mathbf{u}) + \sum_{i=1}^l \lambda_i A_i(\mathbf{u}) \quad (2)$$

where the coefficients λ_i are the appearance parameters. The base (mean) appearance A_0 and appearance images A_i are usually computed by applying Principal Components Analysis to the (shape normalized) training images [7].

Although Equations (1) and (2) describe the shape and appearance variation, they do not describe how to generate a *model instance*. The AAM instance with shape parameters \mathbf{p} and appearance parameters λ_i is created by warping the appearance A from the base mesh \mathbf{s}_0 onto the model shape mesh \mathbf{s} . In particular, the pair of meshes \mathbf{s}_0 and \mathbf{s} define a piecewise affine warp from \mathbf{s}_0 to \mathbf{s} denoted $\mathbf{W}(\mathbf{u}; \mathbf{p})$. Note



Figure 1: AAM shape registration examples across a number of subjects.

that for ease of presentation we have omitted any mention of the 2D similarity transformation that is used with an AAM to normalize the shape [7]. In this paper we include the normalizing warp in $\mathbf{W}(\mathbf{u}; \mathbf{p})$ and the similarity normalization parameters in \mathbf{p} . See [9] for the details of how to do this.

4.2. 3D Active Appearance Models

The *3D shape* $\bar{\mathbf{s}}$ of a 3D AAM [10] is a 3D triangulated mesh. In particular, $\bar{\mathbf{s}}$ is a column vector containing the vertex locations of the mesh. 3D AAMs also allow linear shape variation. The 3D shape vector $\bar{\mathbf{s}}$ can be expressed as a base shape $\bar{\mathbf{s}}_0$ plus a linear combination of \bar{m} shape vectors $\bar{\mathbf{s}}_i$:

$$\bar{\mathbf{s}} = \bar{\mathbf{s}}_0 + \sum_{i=1}^{\bar{m}} \bar{p}_i \bar{\mathbf{s}}_i \quad (3)$$

where the coefficients \bar{p}_i are the shape parameters. 3D AAMs can be computed from training data consisting of a set of *3D range* images with the mesh vertices located in them. In this paper we automatically compute 3D shape from a training set of 2D examples using non-rigid structure-from-motion [10].

The appearance of a 3DMM is a 2D image $A(\mathbf{u})$ just like the appearance of a 2D AAM. The appearance variation of a 3DMM is also governed by Equation (2) and is computed in a similar manner by applying Principal Components Analysis to the unwarped input texture maps.

To generate a 3D AAM *model instance*, an image formation model is needed to convert the 3D shape $\bar{\mathbf{s}}$ into a 2D mesh, onto which the appearance is warped. The following *scaled orthographic* imaging model is used:

$$\mathbf{u} = \mathbf{P}_{\text{so}}(\mathbf{x}) = \sigma \begin{pmatrix} i_x & i_y & i_z \\ j_x & j_y & j_z \end{pmatrix} \mathbf{x} + \begin{pmatrix} o_u \\ o_v \end{pmatrix}. \quad (4)$$

where (o_u, o_v) is an offset to the origin, the projection axes $\mathbf{i} = (i_x, i_y, i_z)$ and $\mathbf{j} = (j_x, j_y, j_z)$ are orthonormal ($\mathbf{i} \cdot \mathbf{i} = \mathbf{j} \cdot \mathbf{j} = 1, \mathbf{i} \cdot \mathbf{j} = 0$), and σ is the scale. The 3D AAM instance is computed by first projecting every 3D shape vertex $\mathbf{x} = (x, y, z)^T$ onto a 2D vertex \mathbf{u} using Equation (4). The appearance $A(\mathbf{u})$ is then warped onto the 2D mesh (taking into account visibility) to generate the final model instance.

4.3. AAM Tracking

We derive features for each of the 20 subjects in the FACS database based on AAM tracking. We use real-time 2D+3D fitting algorithm of [10]. To improve tracking performance and robustness, each subject is tracked using a person-specific AAM model [11].

To compute subject-independent features for use in classification we also build a single *generic* AAM model for all subjects. A final AAM tracking result is obtained by projecting each person-specific tracking result into the generic model. This simply involves computing the 2D shape \mathbf{p} , 3D shape $\bar{\mathbf{p}}$, and appearance parameters λ in the generic model given the vertex locations \mathbf{s} of the person-specific fit.

4.4. AAM Representations

As the AAM allows one to decouple appearance and shape from of a face image a number of representations can be derived (see Figure 2),

2DS: *2D shape, s* representation (see Equation (1)) of the face and its facial features. Normalization processes can be applied such as: (i) the removal of the similarity transform (i.e. remove 2D translation, scale, and rotation), and (ii) the application of an affine transform given some fixed anchor points on the face. We shall

refer to (i) and (ii) as representation **2DS-NS** and **2DS-A** respectively.

2DA: *2D appearance*, $A(\mathbf{u})$ representation (see Equation (2)) of the appearance of the face given a static mean shape.

3DS: *3D shape*, $\bar{\mathbf{s}}$ representation (see Equation (3)) where the position in 3D of the face and its facial features has been estimated. Normalization is applied through the removal of the similarity transform in 3D (i.e. remove 3D translation, scale and rotation).

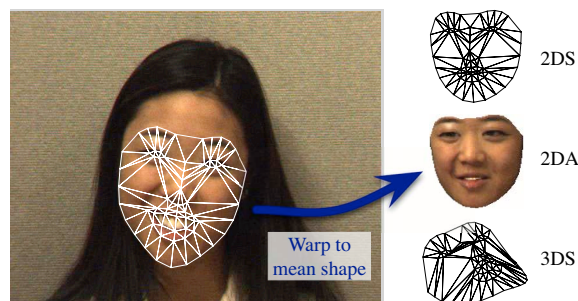


Figure 2: AAM Derived features. The AAM tracked result is used to compute the 2D shape, the shape-free appearance, and the 3D shape.

5. Classifiers

In this paper, since we are dealing with peak-to-peak recognition, we explored two commonly used classifiers for facial action recognition [3, 6].

5.1. Nearest Neighbor (NN)

Nearest neighbor (NN) classifiers are typically employed in scenarios where there are many classes, and there is a minimal amount of training observations for each class (e.g. face recognition); making them well suited for the task of facial action recognition. A NN classifier seeks to find of N labeled train observations $\{\mathbf{o}_i\}_{i=1}^N$ the single closest observation to the unlabeled test observation \mathbf{o}^* ; classifying \mathbf{o}^* as having the nearest neighbor's label.

When N is small the choice of distance metric $D(\mathbf{a}, \mathbf{b})$ between observation points becomes especially important [12]. One of the most common distance metrics employed in face recognition and facial action recognition is the Mahalanobis distance,

$$D(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^T \mathbf{W} (\mathbf{a} - \mathbf{b}) \quad (5)$$

where \mathbf{a} and \mathbf{b} are observation vectors being compared and \mathbf{W} is a weighting matrix. It is often advantageous to

attempt to learn \mathbf{W} from the train-set. Two common approaches to learn \mathbf{W} are,

Principal Component Analysis (PCA): attempts to find the K eigenvectors $\mathbf{V} = \{\mathbf{v}_k\}_{k=1}^K$, corresponding to the K largest eigenvalues, of the train-set's covariance matrix. These K eigenvectors can be thought of as the K largest modes of linear variation in the train-set. The weighting matrix can then be defined as $\mathbf{W} = \mathbf{V}\mathbf{V}^T$. Typically, $K \ll N$ thereby constraining the matching of \mathbf{a} and \mathbf{b} to a subspace where training observations have previously spanned.

Linear Discriminant Analysis (LDA): attempts to find the K eigenvectors $\mathbf{V} = \{\mathbf{v}_k\}_{k=1}^K$ of $\mathbf{S}_b\mathbf{S}_w^{-1}$ where \mathbf{S}_b and \mathbf{S}_w are the within- and between- class scatter matrices of the train-set. These K eigenvectors can be thought of as the K largest modes of discrimination in the train-set. Since $\mathbf{S}_b\mathbf{S}_w^{-1}$ is not symmetrical, we must employ simultaneous diagonalization [12] to find the solution. PCA is typically applied before LDA, especially if the dimensionality of the raw face representations is large, so as to minimize sample-size noise.

If there is not enough training data and many classes LDA overfits and can perform poorly, on the other hand PCA could not be discriminative enough if two classes span a similar subspace. For the purposes of this paper PCA used in conjunction with LDA was found to be most suitable for the task of facial action recognition; we shall refer to this classifier as *NN-LDA*.

5.2. Support Vector Machine (SVM)

Support vector machines (SVMs) have been demonstrated to be extremely useful in a number of pattern recognition tasks including face and facial action recognition. This type of classifier attempts to find the hyper-plane that maximizes the margin between positive and negative observations for a specified class. A linear SVM classification decision is made for an unlabeled test observation \mathbf{o}^* by,

$$\mathbf{w}^T \mathbf{o}^* \begin{cases} \text{true} & \geq b \\ \text{false} & & \end{cases} \quad (6)$$

where \mathbf{w} is the vector normal to the separating hyperplane and b is the bias. Both \mathbf{w} and b were estimated so that they minimize the structural risk of a train-set. Typically, \mathbf{w} is not defined explicitly, but through a linear sum of support vectors. As a result SVMs offer additional appeal as they allow for the employment of non-linear combination functions through the use of kernel functions such as the *radial basis function* (RBF), *polynomial*, *sigmoid* kernels. A RBF kernel was used in our experiments throughout this paper

Feature	Whole		Whole - AU 0		Brow		Brow - AU 0	
	NN-LDA	SVM-RBF	NN-LDA	SVM-RBF	NN-LDA	SVM-RBF	NN-LDA	SVM-RBF
2DS	35.63	20.69	65.52	51.72	28.74	25.29	62.07	64.37
2DS-NS	54.02	60.92	77.01	82.76	56.32	60.92	80.46	89.66
2DS-A	55.17	44.83	67.82	68.97	56.32	59.77	68.97	72.41
2DA	54.02	39.08	59.77	70.11	57.47	42.53	62.07	75.86
3DS	48.28	45.98	57.47	62.07	42.53	47.13	55.17	70.11

Table 1: This table depicts results for feature evaluation on the task of recognition for AUs 1, 1+2 and 4. One can see that the optimal result (italicized) was achieved using a SVM classifier for the 2D shape representation where no similarity transform is applied (i.e. 2DS-NS). The normalization steps of restricting the area of the face analyzed to the eye region, and the normalization by AU 0 also have an additional benefit.

Feature	Whole		Whole - AU 0		Brow		Brow - AU 0	
	NN-LDA	SVM-RBF	NN-LDA	SVM-RBF	NN-LDA	SVM-RBF	NN-LDA	SVM-RBF
2DS	25.24	17.48	54.37	43.69	22.33	21.36	58.25	54.37
2DS-NS	45.63	51.46	59.22	70.87	45.63	53.40	71.84	79.61
2DS-A	45.63	51.46	53.40	60.19	48.54	52.43	60.19	64.08
2DA	53.40	32.04	20.83	36.52	49.51	51.46	64.08	71.84
3DS	30.10	35.92	45.63	52.43	33.98	35.92	47.57	59.22

Table 2: This table depicts results for feature evaluation on the task of recognition for AUs 1, 1+2 4 and 5. One can see that the optimal result (italicized) was achieved using a SVM classifier for the 2D shape representation where no similarity transform is applied (i.e. 2DS-NS). This result is consistent with Table 1.

due to its good performance, and ability to perform well in many pattern recognition tasks [13]. Please refer to [13] for additional information on SVM estimation and kernel selection. We shall refer to this classifier as *SVM-RBF*.

Since SVMs are intrinsically binary classifiers, special steps must be taken to extend them to the multi-class scenario required for facial action recognition. In our work, we adhered to the “one-against-one” approach [13] in which $K(K-1)/2$ classifiers are constructed, where K are the number of AU classes, and each one trains data from two different classes. In classification we use a voting strategy, where each binary classification is considered to be a single vote. A classification decision is achieved by choosing the class with the maximum number of votes.

6. Recognition Experiments

Table 1 depicts the recognition results for our experiments on both the SVM and NN-LDA classifiers for the AU classes 1, 1+2 and 4. A number of AAM derived representations were analyzed. Specifically, we looked at representations of the face based on 2D shape (2DS), 3D shape (3DS) and 2D appearance (2DA). In our experiments we also investigated a number of normalization techniques: (i) using the entire face (*whole*), (ii) using the entire face minus the local neutral (i.e. AU 0) face (*whole-AU0*), (iii) using only the eye/brow region (*brow*), and (iv) using only the

eye/brow region minus the local neutral (i.e. AU 0) face (*brow-AU0*). Normalizations (iii) and (iv) were rationalized based on our prior knowledge of where the AUs were occurring spatially within the face. Similar recognition results can be seen in Table 2 for the harder problem of recognizing AU classes 1, 1+2, 4, and 5.

One can see some immediate trends in Tables 1 and 2. First, there does not seem to be any benefit in employing 3D shape representations of the face, even after the removal of the similarity transform. One hypothesis for this poor result, can be attributed to the noise associated with inferring the depth information of the face. Due to this noise, any 3D normalization, such as the removal of the similarity transform, would add to the already considerable “real-world” within-class variation.

A second trend worth noting is that normalization in the form of: (a) restricting the spatial context of the facial action recognition task (i.e. restricting recognition to the area around the eyes/brow), and (b) subtracting the local AU 0 from each sequence, significantly improves performance for nearly every feature representation investigated. Third, 2D shape is the dominant feature with both the removal of the similarity transform (2DS-NS) and the application of an affine transform (2DS-A) performing well. However, provided one normalizes the 2D appearance (2DA) by the local AU 0, the 2D appearance seems to be quite useful.

In Table 3 we see the confusion matrix for the dominant

		Observed			
		1	1+2	4	5
Actual	1	86.42	11.54	3.85	0.00
	1+2	3.45	96.55	0.00	0.00
	4	12.50	0.00	84.38	3.12
	5	43.75	6.25	18.75	31.25

Table 3: Confusion matrix for the shape feature 2DS-NS, demonstrating good performance on AUs 1, 1+2 and 4, but poor performance on AU 5.

		Observed			
		1	1+2	4	5
Actual	1	76.92	19.23	3.85	0
	1+2	13.79	86.21	0	0
	4	15.62	18.75	62.5	3.12
	5	18.75	12.5	12.5	56.25

Table 4: Confusion matrix for the appearance feature 2DA, demonstrating reasonable performance on AUs 1, 1+2 and 4, but much better performance, with respect to 2DS-NS, on AU 5.

representation 2DS-NS using a SVM-RBF classifier for the task of recognizing AU classes 1, 1+2, 4, and 5. Interestingly the performance of the recognizer suffers mainly from the poor job it does on AU 5.

Inspecting Table 4 however, for the 2DA appearance feature one can see this recognizer does a good job on AUs 1, 1+2 and 4, but does a better job on AU5 than 2DS-NS does. This may indicate that shape and appearance representations of the face may hold some complimentary information with regard to recognizing facial actions.

7. Discussion

In this paper we have explored a number of representations of the face, derived from AAMs, for the purpose of facial action recognition. We have demonstrated that a number of representations derived from the AAM are highly useful for the task of facial action recognition. A number of outcomes came from our experiments,

- It was demonstrated that restricting the face area according to the spatial location of the AU improved recognition performance. This improvement could be attributed to the removal of unwanted AU artifacts in non-essential areas of the face. We have some additional results that speculate different that the shape and appearance representations of the face may be complimentary for different AUs.
- Normalization by the AU 0 frame for *both* shape and texture representations is significantly beneficial for recognition performance.
- Shape features have a large role to play in facial action unit recognition. Based on our initial experiments the

ability to successfully register the shape of the face can be highly beneficial in terms of AU recognition performance.

Some additional work still needs to be done, with model based representations of the face, in obtaining adequate 3D depth information from the face. We believe further improvement in this aspect of model based representations of the face, could play large dividends towards the lofty goal of automatic ubiquitous facial action recognition.

References

- [1] P. Ekman, W. V. Friesen, and J. C. Hager, "Facial action coding system," tech. rep., Research Nexus, Network Research Information, Salt Lake City, UT, 2002.
- [2] J. F. Cohn, Z. Ambadar, and P. Ekman, "Observer-based measurement of facial expression with the Facial Action Coding System," in *The handbook of emotion elicitation and assessment*, Oxford University Press Series in Affective Science.
- [3] G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Classifying Facial Actions," *IEEE Trans. PAMI*, vol. 21, pp. 979–984, October 1999.
- [4] M. Pantic and L. J. M. Rothkrantz, "Toward an Affect-Sensitive Multimodal Human Computer Interaction," *Proceedings of the IEEE*, vol. 91, September 2003.
- [5] T. Cootes, G. Wheeler, K. Walker, and C. Taylor, "Coupled-view active appearance models," in *Proc. of the British Machine Vision Conference*, 2000.
- [6] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, and J. I. Fasel, I.; Movellan, "Recognizing Facial Expression: Machine Learning and Application to Spontaneous Behavior," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 568–573, June 2005.
- [7] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *PAMI*, vol. 23, no. 6, pp. 681–685, 2001.
- [8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York, NY, USA: John Wiley and Sons, Inc., 2nd ed., 2001.
- [9] I. Matthews and S. Baker, "Active Appearance Models revisited," *IJCV*, vol. 60, no. 2, pp. 135–164, 2004.
- [10] J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2d+3d active appearance models," vol. II, (Washington, D.C.), pp. 535–542, June 2004.
- [11] R. Gross, I. Matthews, and S. Baker, "Generic vs. person specific active appearance models," pp. 457–466, Sept. 2004.
- [12] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, 2nd ed., 1990.
- [13] C. W. Hsu, C. C. Chang, and C. J. Lin, "A practical guide to support vector classification," tech. rep., Department of Computer Science, National Taiwan University, 2005.