Combining AAM Coefficients with LGBP histograms in the Multi-Kernel SVM Framework to Detect Facial Action Units.

Thibaud Senechal[1], Vincent Rapp[1,2], Hanan Salam[3], Renaud Seguier[3], Kevin Bailly[4], Lionel Prevost[2]

[1]ISIR
CNRS UMR 7222
Universite Pierre et Marie Curie, Paris
{rapp, senechal}@isir.upmc.fr

[2]LAMIA
EA 4540
University of French West Indies & Guyana
lionel.prevost@univ-ag.fr

[3]Supelec / IETR (UMR 6164)
Avenue de la Boulaie, 35511, Cesson-Sevigne
{salam, seguier}@supelec.fr

[4]Institut Telecom - Telecom ParisTech
CNRS/LTCI, Paris
kevin.bailly@telecom-paristech.fr

***Abstract*— This study presents a combination of geometric and appearance features used to automatically detect Action Units in face images. We use one multi-kernel SVM for each Action Unit we want to detect. The first kernel matrix is computed using Local Gabor Binary Pattern (LGBP) histograms and a histogram intersection kernel. The second kernel matrix is computed from AAM coefficients and a RBF kernel. During the training step, we combine these two type s of features using the recent SimpleMKL algorithm. SVM outputs are then filtered to exploit dynamic relationships between Action Units.**

I. Introduction

A current challenge in designing computerized environments is to place the human user at the core of the system. Traditional computer interfaces ignore users affective states, resulting in a large loss of valuable information for the interaction process. To recognize affective state, human-centered interfaces should interpret gestures, voice and facial movements. Among all these topics, emotions and Action Units (AUs) detection is one of the most active in computer vision. Many systems have been proposed in the literature, but they all suffer of a lack of a common evaluation protocol.

The Facial Expression and Analysis challenge (FERA), organized in conjunction with the IEEE International conference on Face and Gesture Recognition 2011, fits into this scheme: allow comparison between several frameworks.

To do so, it uses a partition of the GEMEP corpus [1]. This dataset consists of recordings of 10 actors displaying a range of expressions, while uttering a meaningless phrase, or the word Aaah. There are 7 subjects in the training data, and 6 subjects in the test set, 3 of which are not present in the training set.

As challenger, we propose here to fusion geometric and appearance features: Local Gabor Binary Pattern (LGBP) histograms and Active Appearance Model (AAM). LGBPs, introduced by Zhang et al. [2] for face recognition, exploit multi-resolution and multi-orientation links between pixels and are very robust to illumination and misalignment. Moreover, the use of histograms results in the loss of spatial information which really depends on identity. One of the drawback would be the inability to capture some subtle movements useful for Action Unit recognitions. To deal with, we decide to look for another set of features in which this information does not lack. So, we choose to use Active Appearance Model (AAM) introduced by Cootes et al. [3]. An AAM contains a statistical model of the shape and grey-level appearance of the face which can generalize to almost any valid example. The AAMs can provide important spatial information of key facial landmarks but are dependent of an accurate matching of the model to the face images.

To perform action unit detection, we select Support Vector Machines for the ability to find an optimal frontier between positive and negative in binary classification problems. As both features (LGBP histograms and AAM coefficients) are very different, we do not concatenate them in a single vector. Instead of using one single kernel function, we decide to use two different kernels, one adapted to LGBP histograms and the other, to AAM coefficients. We combine these kernels in a multi-kernel SVM framework [4]. Finally, to deal with temporal aspects of action unit display, we post-process the classification outputs using filtering and thresholding technics.

The paper is organized as follows. Section II describes image coding and details LGBP histogram and AAM coefficient calculation. Section III explains classification process to detect AUs in facial images and post-processing temporal analysis. Section IV details training and validation processes within two setup: namely person-independent and person-specific. Section V reports AUs detection results on the GEMEP-FERA test dataset. Finally section VI concludes the paper.

II. Features

A. LGPB histograms

In this section, we describe how we compute Local Gabor Binary Patterns histograms from facial images (Fig. 1). To

pre-process data, we automatically detect eyes using our own feature localizer [5]. Eyes localization is used to remove variations in scale, position and in-plane rotation. We obtain facial images with the same size of $128 * 128$ pixels and eye centers always at the same coordinates.

1) Gabor magnitude pictures: The Gabor magnitude pictures are obtained by convolving facial images with Gabor filters :

$$G_{\mathbf{k}}(\mathbf{z}) = \frac{\mathbf{k}^2}{\sigma^2} e^{(-\frac{\mathbf{k}^2}{2\sigma^2} z^2)} (e^{i\mathbf{kz}} - e^{-\frac{\sigma^2}{2}}) \quad (1)$$

Where $\mathbf{k} = k_v e^{i\phi_u}$ is the characteristic wave vector. We use three spatial frequencies $k_v = (\frac{\pi}{2}, \frac{\pi}{4}, \frac{\pi}{8})$ and six orientations $\phi_u = (\frac{k\pi}{6}, k \in \{0 \dots 5\})$ for a total of 18 Gabor filters. As the phase is very sensitive, only the magnitude is generally kept. It results in 18 Gabor magnitude pictures.

2) Local Binary Pattern (LBP): The LBP operator was first introduced by [6]. It codes each pixel of an image by thresholding its 3×3 neighborhood with its value and considering the result as a binary number. The LBP of a pixel $\mathbf{p}$ (value $f_{\mathbf{p}}$) with a neighborhood $\{f_k,\ k = 0...7\}$ is defined as:

$$LBP(\mathbf{p_c}) = \sum_{k=0}^{7} \delta(f_k - f_p) 2^k \quad (2)$$

where

$$\delta(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \quad (3)$$

3) Local Gabor Binary Pattern (LGBP): We apply the LBP operator on the 18 Gabor magnitude pictures resulting in 18 LGBP-maps per facial image. This combination of the Local Binary Pattern operator with Gabor wavelets exploits multi-resolution and multi-orientation links between pixels. This has been proven to be very robust to illumination changes and misalignments [2].

4) Histogram sequences: Each area of the face contains different useful information for AU detection. Thus, we choose to divide the face in many areas and compute one histogram per area. Such a process is generally applied in histogram-based methods for object classification. We divide each facial image into $4 \times 4 = 16$ non-overlapping regions. This cutting has been chosen because some AUs modify a part of the face as small as these regions. For example, AU1 only modifies the brow inner corners.

$$\mathbf{h}(k) = \sum_{p} I(k \leq f_p < k+1), \quad k = 0 \cdots 255 \quad (4)$$

$$I(A) = \begin{cases} 1 & \text{if } A \text{ is true} \\ 0 & \text{if } A \text{ is false} \end{cases} \quad (5)$$

For each face i, we get one vector H_i by concatenating all the histograms. H_i is the concatenation of $16 \times 6 \times 3 = 288$ histograms computed on each region, orientation and spatial frequency resulting in $288 \times 256 = 73728$ features per facial image.

5) Reducing histogram bins number: Ojala et al.[6] showed that a small subset of the patterns accounted for the majority of the texture of images. They only keep the uniform patterns, containing at most two bitwise transitions from 0 to 1 for a circular binary string.

To reduce the number of bins per histogram, we choose a slightly different approach. As we want the contribution of all patterns, we decide to group the occurrence of different patterns into the same bin. First, we only keep patterns that do not contain any isolated 0 or 1, i.e. 0 or 1 surrounded by ones or zeroes respectively for a circular binary string. For example we keep 00011000 or 10011001 and not 1000 0000 or 1101 1001. There are 30 patterns with this property. We choose these patterns because they are the most useful (in our facial images database, 70% of the patterns were one of these 30 within 256 possibilities) and are close from one bit to almost all other patterns. Then, patterns are grouped with their closest neighbor within these 30 patterns. When a pattern has more than one closest neighbor, its contribution is equally divided between all its neighbors. For example, one third of the number of pattern 0000 0001 is added to the bin represented by the pattern 0000 0000, the bin represented by 0000 0011 and the one represented by 1000 0001.

It finally results in 30 bins per histogram instead of 256 and the histogram H_i coding the face is a histogram of $288 \times 30 = 8640$ bins.

B. 2.5D Active Appearance Model

We have used a 2.5D AAM [7] which is constructed by:

- 2D landmarks of frontal and profile views of the facial images combined to make 3D shape
- 2D textures of frontal view of the facial images, mapped on the mean 3D shape.

During the training phase of 2.5D AAM, 68 points are marked manually on each facial image.

Mean of these 3D landmarks, which is called the mean shape $\bar{s}$, is calculated. This 3D mean shape extracts the frontal views of all the textures of face images. Then, we calculate the mean of these textures $\bar{g}$. A Principal Component Analysis (PCA) is performed on these shapes and textures to obtain shape parameters b_s and texture parameters b_g with 95% of the variation stored in these parameters. Hence shape and texture are synthesized by $s_i = \bar{s} + \phi_s * b_s$ and $g_i = \bar{g} + \phi_g * b_g$ respectively. Matrices ϕ_s, ϕ_g contain eigenvectors of variation in shapes and textures. Parameters b_s, b_g are then combined as $b = [b_s b_g]^T$ and a final PCA is performed to obtain the appearance parameters by the equation $b = \phi_c * C$, where ϕ_c are the eigenvectors and C is the matrix of the appearance parameters, which are used to render the shape and texture of each face of the database.

This 2.5D AAM can be translated as well as rotated with the help of pose vector given as:

$$P = [\theta_{pitch}, \theta_{yaw}, \theta_{roll}, t_x, t_y, Scale]^T \quad (6)$$

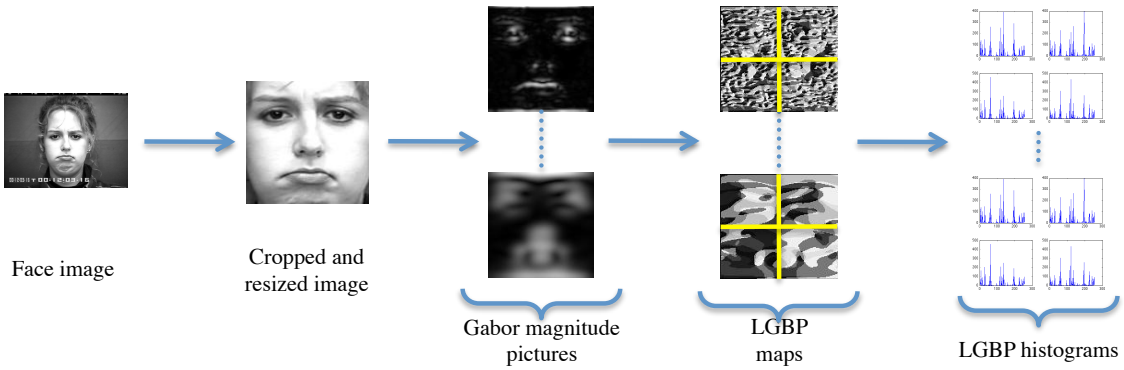

Fig. 1. Flowchart: Local Gabor Binary Pattern histograms computation.

III. Classifiers

To perform the AU recognition task, we have to make several binary decisions. Hence, we chose to train one Support Vector Machine (SVM) per AU. To train one SVM, all images containing the specific AU are used as positive samples (target class) and the other images are used as negatives (non-target class).

A. Multi-kernels SVMs

Given training samples composed of LGBP histograms and an AAM appearance vector $x_i = (H_i C_i)$, associated with labels y_i (target or non-target), the classification function of the SVM associates a score s to the new sample $x = (H, C)$:

$$s = \left(\sum_{i=1}^{m} \alpha_i k(x_i, x) + b \right) \tag{7}$$

With α_i the dual representation of the hyperplane's normal vector [8]. k is the kernel function resulting from the dot product in a transformed high-dimensional feature space.

In case of multi-kernel SVM, the kernel k can be any convex combination of semi-definite functions.

$$k(x_i, x) = \sum_{j=1}^{K} \beta_j k_j \text{ with } \beta_j \geq 0, \sum_{j=1}^{K} \beta_j = 1 \tag{8}$$

In our case, we have one kernel function per type of features.

$$k = \beta_1 k_{LGBP}(H_i, H) + \beta_2 k_{AAM}(C_i, C) \tag{9}$$

Weights α_i and β_j are set to have an optimum hyperplane in the feature space induced by k. This hyperplane separates positive and negative classes and maximizes the margin (minimum distance of one sample to the hyperplane). This optimization problem has proven to be jointly-convex in α_i and β_j [9], therefore there is a unique global minimum and can be found efficiently.

β_1 represents the weight accorded to the LGBP features and β_2 is the one for the AAM appearance vector. Thus, using a learning database, the system is able to find the best combination of these two types of feature that maximizes the margin.

This is a new way of using multi-kernel learning, instead of combining different kind of kernel functions (for example Gaussian radial basis functions with polynomial functions), we combine different features. The AAMs modeling approach of takes into account the localization of the facial feature points and leads to a shape-free texture less-dependent to identity. But one of the severe drawbacks is the need of a good accuracy for the localization of the facial feature points. As the images of the GEMEP-FERA databases contain high display of expression, this is not always the case. In such cases, multi-kernel SVMs will decrease the weight accorded to AAM, and will only focus on LGBP features which are more robust.

B. Kernel functions

In our previous study [10], we showed that in histogram-based AU recognition, LGBP histograms are well-suited with a Histogram Intersection Kernel:

$$K_{LGBP}(H_i, H_j) = \sum_{k} min(H_i(k), H_j(k)) \tag{10}$$

For the AAM appearance vectors, we use Radial Basis Function (RBF) kernel :

$$K_{AAM}(C_i, C_j) = e^{-\frac{\|s_i - s_j\|_2^2}{2\sigma^2}} \tag{11}$$

With σ a hyper-parameter we have to tune on a cross-validation database.

C. Temporal filtering

To include temporal information, we apply, for each AU, an average filter to the outputs of each SVM classifier of successive frames. The size of the average filter has been set to maximize the F1-measure reached on the training database.

IV. Cross-Validation

In this section we explain the training process of the AUs detector and report experiments on the GEMEP-FERA training dataset . To train AAMs, 400 images from the Cohn-Kanade database and 257 images from the GEMEP-FERA training dataset have been used.

To train SVM classifiers we select from the GEMEP-FERA training dataset around 600 images, one frame from every video for every AU combination present. We have added 486 images from the Cohn-Kanade database [11] (the last image of each sequence) and 400 images from the Bosphorus database [12] all containing at least one of the AU we have to detect for the challenge.

To evaluate the impact of the combination of different features, we have made three experiments. The first one only uses LGBP histograms and one single Histogram Intersection kernel SVM per AU. The second one only uses appearance vector and single Gaussian kernel SVM per AU. Finally, in the third experiment, we combine both information using one multi-kernel SVM per AU. All SVMs are trained using a Matlab implementation of the SimpleMKL algorithm [4].

Two different setups are used:

- a 7-fold subject independent cross-validation. All the images of one subject from the GEMEP-FERA training dataset are used as test set (around 800 images). Images of other subjects within the 600 selected from the GEMEP-FERA training dataset, images from Cohn-Kanade and images from Bosphorus are used as training set (around 1400 images)
- a 2-fold person-specific cross-validation. One half of the GEMEP-FERA images from all subjects are used as test set (around 2500 images). The other half is merged with Cohn-Kanade and Bosphorus images to train the system.

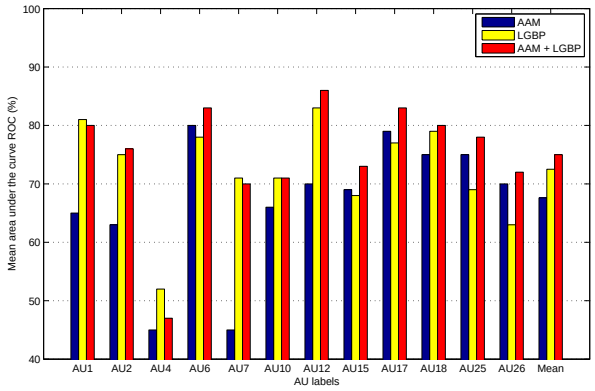

Fig. 2. Results in a person-independent setup

We report in fig. 2 and fig. 3 area under the curve ROC achieved in person-independent and person-specific setup. Optimal values of RBF kernel parameter and SVM slack variables are tuned on the cross-validation set.

First and second experiments compare single-kernel SVM based on LGBP histograms or AAMs. We notice the same

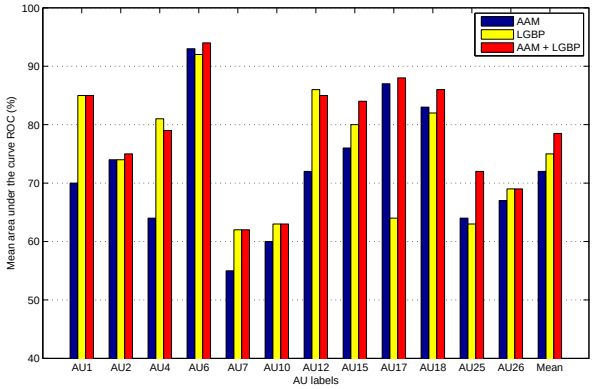

Fig. 3. Results in a person-specific setup

tendencies with person-independent and person-specific setups. LGBP leads to better results than AAMs, especially for lower face action units. The localization of eyebrows by the AAMs is often inaccurate, resulting in worse results for AU1, AU2 and AU4. In the other hand, the localization of the lips being accurate, AAMs permit better results for some lower AUs, like AU25 and AU26 in the person-independent setup and AU17 (chin raiser) in both setups (fig.4).

For the third experiment, multi-kernel SVMs have better or similar performances than single-kernel SVMs. In the case AAMs lead to worse results than LGBPs (AU1 2 7 12), the multi-kernel SVMs perform as well as the single-kernel SVMs using only LGBP. This shows one of the interests of the multi-kernel learning. It is robust, even with features that do not exhibit good generalization performances. When LGBPs and AAMs lead to similar results, their combination increases detection performances.

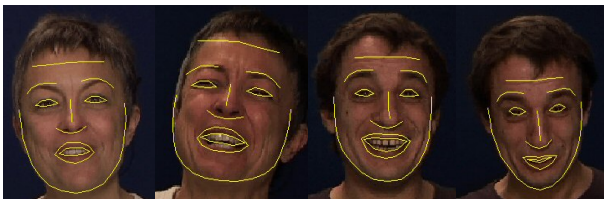

Fig. 4. AAM results on test images highlighting accurate localizations on mouth or chin and localization problems on eyebrows

Finally to learn the final system we proceed as follows:

- we merge the results of the 7-fold subject independent cross-validation. This leads to a SVM output for each AU and for each frame of the GEMEP-FERA training dataset
- for each AU, we learn the size of the average filter and the decision threshold that lead to the best F1-measure.
- we train multi-kernel SVMs on all images using the optimal parameter values found during the 7-fold subject independent cross-validation.

We apply the multi-kernel SVMs on the GEMEP-FERA test dataset to have a score per AU and per frame. Then, we use the average filter and threshold the value to make a binary decision.

AUs	PI		PS		O		
	base	ours	base	ours	naive	base	ours
AU1	0.633	0.809	0.362	0.507	0.506	0.567	0.744
AU2	0.675	0.731	0.400	0.649	0.477	0.589	0.706
AU4	0.133	0.582	0.298	0.680	0.567	0.192	0.610
AU6	0.536	0.834	0.255	0.649	0.626	0.463	0.787
AU7	0.493	0.702	0.481	0.655	0.619	0.489	0.684
AU10	0.445	0.475	0.526	0.477	0.495	0.479	0.476
AU12	0.769	0.803	0.688	0.784	0.739	0.742	0.796
AU15	0.082	0.245	0.199	0.334	0.182	0.133	0.292
AU17	0.378	0.556	0.349	0.432	0.388	0.369	0.535
AU18	0.126	0.431	0.240	0.406	0.223	0.176	0.424
AU25	0.796	0.850	0.809	0.826	0.825	0.802	0.839
AU26	0.371	0.576	0.474	0.508	0.495	0.415	0.546
Avg	0.453	0.633	0.423	0.576	0.512	0.451	0.620

TABLE I

F1 MEASURE FOR AU DETECTION RESULTS ON THE TEST SET FOR THE BASELINE METHOD (BASE), THE NAIVE METHOD AND OUR METHOD. RESULTS ARE SHOWN FOR THE PERSON INDEPENDENT (PI), PERSON SPECIFIC (PS) AND OVERALL (O) PARTITIONS

V. RESULTS

The test data for AU detection consists of 71 videos, of the same kind as the training data videos. Half of the subjects in the test data also appear in the training data, while the other half does not. This way it is possible to assess how well systems generalize to unseen subjects. The test data was pre-processed as described in II

AUs	PI		PS		O	
	base	ours	base	ours	base	ours
AU1	0.845	0.923	0.613	0.795	0.790	0.879
AU2	0.818	0.897	0.640	0.843	0.767	0.880
AU4	0.481	0.651	0.607	0.850	0.526	0.712
AU6	0.690	0.894	0.568	0.825	0.657	0.876
AU7	0.572	0.799	0.530	0.700	0.556	0.754
AU10	0.577	0.492	0.627	0.574	0.597	0.521
AU12	0.738	0.830	0.700	0.839	0.724	0.826
AU15	0.555	0.646	0.567	0.734	0.563	0.690
AU17	0.679	0.735	0.661	0.805	0.646	0.758
AU18	0.620	0.818	0.599	0.746	0.610	0.790
AU25	0.544	0.757	0.669	0.623	0.593	0.700
AU26	0.457	0.712	0.555	0.683	0.500	0.633
Avg	0.631	0.763	0.61	0.751	0.628	0.752

TABLE II

AREA UNDER ROC CURVE FOR AU DETECTION RESULTS ON THE TEST SET FOR THE BASELINE METHOD (BASE), THE NAIVE METHOD AND OUR METHOD. RESULTS ARE SHOWN FOR THE PERSON INDEPENDENT (PI), PERSON SPECIFIC (PS) AND OVERALL (O) PARTITIONS

The F1-measure considers both the precision p and the recall r of the test to compute the score: p is the number of correct results divided by the number of all returned results and r is the number of correct results divided by the number of results that should have been returned. The F1-measure can be interpreted as a weighted average of the precision and recall, where an F1-measure reaches its best value at 1 and worst score at 0.

$$F = 2 \cdot \frac{p \cdot r}{p + r} \tag{12}$$

Table I and table II show scores computed in terms of F1-measure and area under ROC curve for person independent, person specific and overall. They compare results for the baseline method provided by Valstar et al [13] using Local Binary Patterns as features and Support Vector Machines with Radial Basis Function kernels to classify the data. Results from a naive system (setting all predictions of every AU to true) are also given. The overall score is obtained by computing the average over all 12 AUs. These tables show that our system performs well over the baseline and the naive methods.

VI. CONCLUSION

We have proposed here an original framework to perform action unit detection. We combine spatial-independent feature extraction (LGBP histograms) and statistical spatial shape and texture information (AAM coefficients). To deal with these two kinds of information and take the best of both, we propose to use advanced learning machine algorithms. Multi-kernel SVMs can help in selecting the most accurate information as each kernel function is weighted depending on this latter. Cross-validation process in person-independent and person-specific setups shows that combining these information improve overall results. Preliminary test results on non-labeled sequences are promising.

REFERENCES

[1] T. Banziger and K. R. Scherer, "Using actor portrayals to systematically study multimodal emotion expression: The gemep corpus," p. 476, 2007.

[2] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang, "Lgbphs: A novel non-statistical model for face representation and recognition," in *Proc. IEEE Int'l Conf. on Computer Vision (ICCV '05)*, 2005.

[3] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *Proc. IEEE European Conference on Computer Vision (ECCV '98)*, p. 484, 1998.

[4] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *Journal of Machine Learning Research*, vol. 9, pp. 2491–2521, 2008.

[5] V. Rapp, T. Senechal, K. Bailly, and L. Prevost, "Multiple kernel learning svm and statistical validation for facial landmark detection," *IEEE Int'l. Conf. Face and Gesture Recognition (FG'11)*, March 2011 (accepted for publication).

[6] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[7] A. Sattar, Y. Aidarous, and R. Seguier, "Gagm-aam: a genetic optimization with gaussian mixtures for active appearance models," in *IEEE International Conference on Image Processing. (ICIP'08).*, 2008, pp. 3220–3223.

[8] B. Scholkopf and A. Smola, *Learning with Kernels*. MIT Presss, 2002.

[9] G. Lanckriet, N. Cristianini, P. Bartlett, L. Ghaoui, and M. Jordan, "Learning the kernel matrix with semidefinite programming," *The Journal of Machine Learning Research*, vol. 5, p. 27, 2004.

[10] T. Senechal, K. Bailly, and L. Prevost, "Automatic facial action detection using histogram variation between emotional states," *Proc. Int'l Conf. Pattern Recognition (ICPR'10)*, 2010.

[11] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition (AFGR '00)*, 2000, pp. 46–53.

[12] A. Savran, N. Alyuz, H. Dibeklioglu, O. Celiktutan, B. Gokberk, B. Sankur, and L. Akarun, "Bosphorus database for 3d face analysis," *Biometrics and Identity Management*, pp. 47–56, 2008.

[13] M. Valstar, B. Jiang, M. Méhu, M. Pantic, and K. Scherer, "The first facial expression recognition and analysis challenge," *IEEE Int'l. Conf. Face and Gesture Recognition (FG'11)*, 2011.