Real-time View-based Face Alignment using Active Wavelet Networks

Changbo Hu, Rogerio Feris, Matthew Turk
Computer Science Department
University of California
Santa Barbara, CA, 93106
{cbhu,rferis,mturk}@cs.ucsb.edu

Abstract

The Active Wavelet Network (AWN) [9] approach was recently proposed for automatic face alignment, showing advantages over Active Appearance Models (AAM), such as more robustness against partial occlusions and illumination changes. In this paper, we (1) extend the AWN method to a view-based approach, (2) verify the robustness of our algorithm with respect to unseen views in a large dataset and (3) show that using only nine wavelets, our method yields similar performance to state-of-the-art face alignment systems, with a significant enhancement in terms of speed. After optimization, our system requires only 3ms per iteration on a 1.6GHz Pentium IV. We show applications in face alignment for recognition and real-time facial feature tracking under large pose variations.

1. Introduction

This paper addresses the problem of automatic face alignment, which consists of extracting facial feature points of a given face image, so that it can be aligned with a canonical face image. Such task is pre-requisite for many computer vision problems, such as face recognition, facial expression analysis and face tracking.

Extensive research has been conducted on this topic, using methods based on Active Contours [10], Gabor wavelets [17, 11] and deformable models [18], to mention just a few. Among model-based approaches, Active Shape Models (ASM) [3] and Active Appearance Models (AAM) [5] have achieve good results in face alignment.

The ASM method detects facial landmarks through a local-based search constrained by a global shape model, statistically learned from training data. The AAM algorithm elegantly combines shape and texture models, assuming a linear relationship between appearance and pose variation. We refer to the work of Cootes et al.[4] for a comparison between these two methods.

Several variations of AAM have also been proposed to improve the original algorithm, namely view-based AAM [6], Direct Appearance Models [8], a compositional approach [2] and 3D AAM [1]. Despite the success of these methods, problems still remain to be solved. For example, AAM is sensitive to illumination changes, especially if the lighting in the test image significantly differs from the lighting encoded in the training set. Moreover, under the presence of partial occlusion, the PCA-based texture model of AAM causes the reconstruction error to be globally spread over the image, thus impairing alignment.

Recently, we have proposed a new method, called Active Wavelet Networks (AWN) [9], in which a Gabor wavelet network representation (GWN) [11] is used to model the texture variation in the training set. The GWN approach represents a face image through a linear combination of 2D Gabor functions whose parameters (position, scale and orientation) and weights are optimally determined to preserve the maximum image information for a chosen number of wavelets.

Because of the localization property of wavelets, when

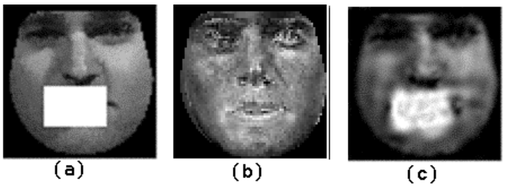


Figure 1: (a) Partial occluded image. (b) PCA reconstruction. Note that the error is spread over the image. (c) Wavelet reconstruction.

partial occlusion or highlight illumination problems arise, the matching is more robust than with AAM. Figure 1 illustrates a comparison between PCA and a Gabor wavelet reconstruction for a partial occluded face image. Note that the error is globally spread over the image in PCA, whereas it remains local in the wavelet representation.

In this paper, we improve our work in the following aspects:

- We extend the AWN method to a view-based approach, allowing robust facial feature tracking under large pose variations.
- We verify the robustness of the AWN algorithm with respect to unseen faces in a large dataset. A real-time, fully automatic face alignment system is presented, with evaluation in FERET database.
- We demonstrate that using only nine wavelets, our method yields accuracy similar to state-of-the-art face alignment systems [5, 3], while posing a significant enhancement in terms of speed. Implemented on a conventional desktop computer, the AWN algorithm requires only 3ms per iteration. In general, given a good initialization, at most ten iterations are sufficient for good convergence.

The remainder of this paper is organized as follows. In Section 2, we present the AWN approach for face alignment, whereas Section 3 describes the extension to a view-based approach. Section 4 covers our experimental results and Section 5 concludes the paper with final remarks and future work.

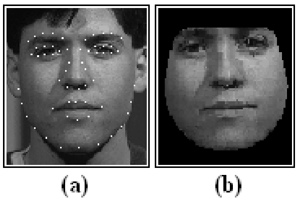


Figure 2: (a) Labelled training image. (b) Shape-free texture.

2. Active Wavelet Networks

In this section, we introduce active wavelet networks for face alignment. Our method starts with a training set, in which each image is labelled with landmark points on the subject's face. Thus, each sample has a labelled shape and an image texture.

Consider the training set of shape and texture to be $\Omega = \{(\mathbf{x}_i, \mathbf{g}_i^x)\}, i = 1...N$,where N is the number of training images, $\mathbf{x}_i = \{(x_j^i, y_j^i)\}, j = 1...M$, is a shape specified by a set of M points, and $\mathbf{g}_i^x$ is the texture enclosed by the shape $\mathbf{x}_i$. We model the shape variation by PCA, and the texture is represented by a GWN model. We will describe the shape model and the GWN texture representation in the following subsections.

2.1. Statistical Shape Model

Given the training set, all shapes are aligned to a common coordinate frame and then the shape variation can be modelled by PCA in a lower dimensional shape space. So, a normalized shape **x** can be approximated as:

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b} \qquad (1)$$

where $\bar{\mathbf{x}}$ is the mean shape, **P** is a set of orthogonal modes of variation and **b** is a set of shape parameters. Using the shape landmarks as control points, we can warp the training images to the mean shape. Figure 2 illustrates a labelled image and its texture warped into the mean shape. The set of shape-free textures $G = \{\mathbf{g}_i^{\bar{x}}\}, i = 1...N$ is used to learn the GWN representation, as described next.

2.2. Wavelet Network Model

We have used a wavelet network to model the face texture as an alternative to Principal Component Analysis in standard AAM. As already mentioned, the use of spatially localized wavelets allows more robustness with respect to partial occlusions and illumination changes.

The constituents of a wavelet network are single wavelets and their associated coefficients. We adopted the odd-Gabor function as the mother wavelet. It is well known that Gabor filters are recognized as good feature detectors and provide the best trade-off between spatial and frequency resolution [14]. Considering the 2D image case, each single odd Gabor wavelet can be expressed as follows:

$$\begin{aligned}\psi_{\mathbf{n}}(\mathbf{x}) &= \exp\left[-\frac{1}{2}(\mathbf{S}(\mathbf{x}-\mu))^T(\mathbf{S}(\mathbf{x}-\mu))\right] \\ &\quad \times \sin\left[(\mathbf{S}(\mathbf{x}-\mu))^T \begin{pmatrix} 1 \\ 0 \end{pmatrix}\right] \end{aligned} \qquad (2)$$

where $\mathbf{x}$ represents image coordinates and $\mathbf{n} = (s^x, s^y, \theta, \mu^x, \mu^y)$ are parameters which compose the terms $\mathbf{S} = \begin{pmatrix} s^x \cos\theta & -s^y \sin\theta \\ s^x \sin\theta & s^y \cos\theta \end{pmatrix}$, and $\mu = \begin{pmatrix} \mu^x \\ \mu^y \end{pmatrix}$, that allow scaling, orientation, and translation. A Gabor wavelet network for a given image consists in a set of n wavelets $\{\psi_{\mathbf{n}_k}\}$ and a set of associated weights $\{w_k\}$, specifically chosen so that the GWN reconstruction:

$$\hat{I}(\mathbf{x}) = \sum_{k=1}^{n} w_k \psi_{\mathbf{n}_k}(\mathbf{x}) \qquad (3)$$

best approximates the target image. We modified the original formulation of GWNs to allow the optimization of a single GWN in a set of shape-free images, obtained through warping, as described in previous section.

2.2.1 Calculation of Wavelet Parameters

Assuming that we have a set of shape-free face images of different people, $\{\mathbf{g}_i^{\bar{x}}\}, 1 \le i \le N$, that are truncated to the region that the face occupies, we calculate the GWN representation parameters as follows:

1. Randomly drop n wavelets of assorted position, scale, and orientation, within the bounds of the normalized face images.

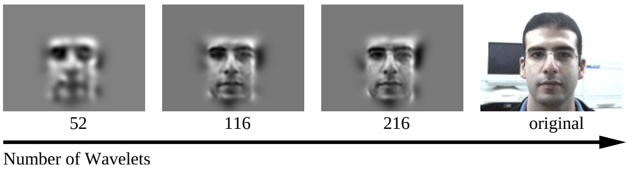


Figure 3: The image shows a facial reconstructions with variable accuracy, considering (from left to right) 52, 116 and 216 wavelets.

2. Perform gradient descent (via Levenberg-Marquardt optimization) over the set of wavelet parameters to minimize the total sum of differences between the training images and their wavelet reconstructions:

$$\arg\min_{\mathbf{n}_k, w_{ik}} \left\| \sum_{i=1}^{N} (\mathbf{g}_i^{\bar{x}} - \sum_{k=1}^{n} w_{ik}\psi_{\mathbf{n}_k}(\mathbf{x})) \right\|^2 \qquad (4)$$

One advantage of the GWN approach is that one can trade-off computational effort with representational accuracy, by increasing or decreasing the number n of wavelets (see Figure 3).

2.2.2 Calculation of Texture Parameters

In the standard Shape-AAM method, the texture parameters for a given image are computed by projecting the image into an eigenspace learned from the training set. In our method, the texture parameters $\{t_k\}, k = 1...n$ correspond to wavelet coefficients, obtained by projecting the image into the learned wavelet subspace.

For an orthogonal wavelet basis, these coefficients may be calculated by simple inner products of the image with each wavelet function $\psi_{\mathbf{n}_k}$, which guarantees an optimal image reconstruction (in the least square sense). However, Gabor functions are not orthogonal, so the texture parameter cannot be computed by inner products of the image with the wavelet functions. In this case, we need to consider a family of dual wavelets $\tilde{\Psi} = \{\tilde{\psi}_{\mathbf{n}_1} \ldots \tilde{\psi}_{\mathbf{n}_n}\}$ [7] to obtain the set of coefficients (or texture parameters) that leads to an optimal image reconstruction. The wavelet $\tilde{\psi}_{\mathbf{n}_j}$ is the dual wavelet of the wavelet $\psi_{\mathbf{n}_i}$ iff $\langle \psi_{\mathbf{n}_i}, \tilde{\psi}_{\mathbf{n}_j} \rangle = \delta_{i,j}$.

Given a normalized face image **g** and a set of optimized wavelets $\Psi = \{\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N}\}$, the texture parameters are given by:

$$t_k = \langle \mathbf{g}, \tilde{\psi}_{\mathbf{n}_k} \rangle. \quad (5)$$

It can be shown that $\tilde{\psi}_{\mathbf{n}_k} = \sum_l \left(A^{-1}\right)_{k,l} \psi_{\mathbf{n}_l}$, where A is the wavelet interference matrix, with $A_{k,l} = \langle \psi_{\mathbf{n}_k}, \psi_{\mathbf{n}_l} \rangle$.

It is important to mention that we can associate each wavelet and its respective dual function with lookup tables, which are computed beforehand. Such tables store only the values of the wavelets within their spatial support, significantly increasing efficiency in subspace projection and reconstruction.

2.3. AWN Search

Given a new face image, and a rough estimation of face pose, the search process aims to determine shape and pose parameters that best fit the model into the new image. The AWN search algorithm is a variation of the Shape-AAM method, where the main difference is the calculation of texture parameters and image reconstruction, which are based on the GWN model.

Let $\mathbf{g}^{\bar{x}}$ be the normalized image enclosed by a shape $\mathbf{x}$ and $\hat{\mathbf{g}}^{\bar{x}}$ its GWN reconstruction. The residual between both images is:

$$\delta \mathbf{g} = \mathbf{g}^{\bar{x}} - \hat{\mathbf{g}}^{\bar{x}} \quad (6)$$

The residual δg is used to drive the shape parameters $\mathbf{b}$ and the affine pose parameters $\mathbf{p}$, assuming a linear relationship:

$$\delta \mathbf{b} = \mathbf{B} \delta \mathbf{g}, \quad \delta \mathbf{p} = \mathbf{P} \delta \mathbf{g} \quad (7)$$

where the two regression matrices $\mathbf{B}$ and $\mathbf{P}$ are computed offline, by perturbing the face model parameters on training data. Our search algorithm can be described as follows.

Given a new face image, a starting shape $\mathbf{x}$ and pose $\mathbf{p}$,

1. Sample the image enclosed by the current shape and normalize it to obtain $\mathbf{g}^{\bar{x}}$
2. Use GWN to compute texture parameters using eq. 5 and reconstruct the texture $\hat{\mathbf{g}}^{\bar{x}} = \sum_{k=1}^{n} t_k \psi_{\mathbf{n}_k}$.
3. Compute the residual using eq. 6
4. Predict the the shape and the pose parameters using eq. 7.
5. If the change of $\delta \mathbf{g}$ is small enough or the maximum number of iterations was reached, stop; else go to step 1.

A successful search results in a AWN model that is well aligned with the input face image.

3. View-based AWN

In this section, we extend the AWN algorithm to a view-based approach, so that it is capable of handling significant face pose changes.

View-based approaches [6, 13] have been successfully applied in many computer vision problems. Rather than relying on an explicit 3D model, a set of 2D models corresponding to different views are used to handle pose variations.

We extend the AWN method, discussed in the previous section, to allow face alignment under different views. Basically, the whole range of views, from profile to profile, is partitioned into several subranges, and one AWN model is trained to represent the shape and texture of each subrange.

Specifically, we use 5 view ranges: $[-90^o, -50^o]$, $[-55^o, -10^o]$, $[-15^o, 15^o]$, $[10^o, 55^o]$ and $[50^o, 90^o]$ with 0^o being the frontal view. Given a set of training images in each subrange, each AWN model is trained as described in Sections 2.1 and 2.2.

Due to symmetry, we just need to train 3 models for the ranges $[-15^o, 15^o]$, $[10^o, 55^o]$ and $[50^o, 90^o]$. Figure 4 illustrates some labelled training samples for the considered views, with their respective shape-free texture.

View-based alignment in a new face image is achieved by (1) selecting the appropriate view model and (2) running AWN search as described in Section 2.3. The first task may be accomplished in several ways. For example, a pose estimator [12] could be used to select the view range. Another way is to run the AWN search for the view models in parallel and select the best match.

In our implementation, model selection is achieved during tracking. In the initial frame, we assume we have a frontal view, and the AWN model corresponding to the view range $[-15^o, 15^o]$ is used. Then, a feature-based head pose

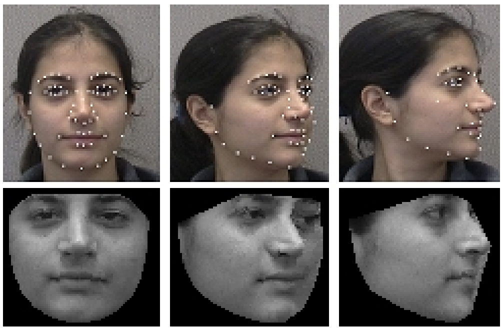


Figure 4: Examples of labelled training samples in different view subranges, with respective shape-free normalized faces.

estimation [15] is used to select the model for the next video frame.

Our head pose estimator uses a subset of the shape feature points, located by the AWN algorithm, to determine the 3D transformation from a face reference frame to the camera reference frame. This is achieved by finding a least-squares fit between a generic 3D model of facial features and the tracked features, under weak perspective (see [15] for more details). Note that the view subranges overlap, so that a rough head pose estimation is sufficient for selecting the appropriate model.

4. Experimental Results

We start by showing that the AWN method may run extremely fast , while providing results as accurate as the Active Appearance Model (AAM) algorithm.

We have learned the AWN and AAM models in a training set containing 40 face images of different individuals. We derived a statistical shape model using a 15-dimensional eigenspace, capturing 98% of variation in the training set. The texture model for AAM was built using 75 modes, also capturing 98% of the total texture variation.

Using only nine wavelets, the AWN method yields similar performance to AAM in a small test database of 90 images (128x192 pixels) of different individuals, under normal conditions. Figure 5 (top) shows the average error per shape point for both methods for each image in the test set. The range of initial location for good convergence is about 8 pixels for both methods. The initial model location was randomly chosen with maximum range of 8 pixels from the ground-truth.

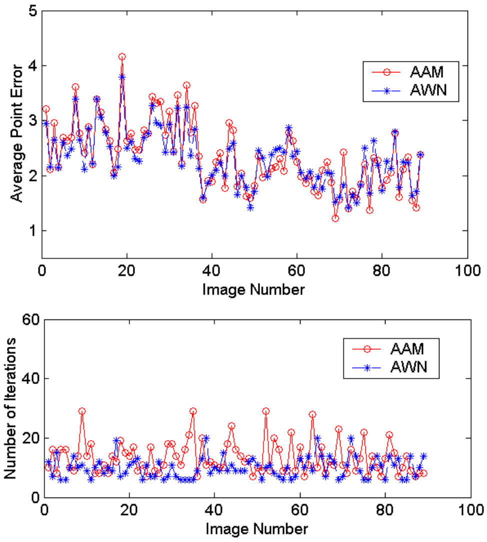


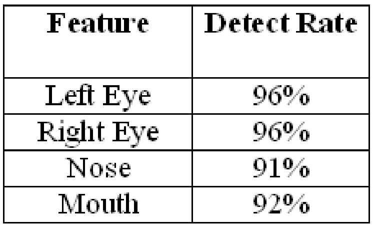
Figure 5: Top: Comparison of accuracy of AWN with nine wavelets and AAM. The two methods achieve similar performance. Bottom: number of iterations required for each image.

Figure 5 (bottom) shows the number of iterations for AWN and AAM in each test image, where AWN offers a slight advantage over AAM. After code optimization, our method requires only 3ms per iteration on a Pentium IV 1.6 GHz, with 512MB of RAM.

We also verified the robustness of the AWN algorithm with respect to unseen faces in a large dataset. More than 3000 frontal view face images of FERET database were used to evaluate a *real-time*, fully automatic face alignment system for face recognition.

The system is composed of a face detector module, recently proposed by Viola and Jones [16], which provides initialization for our AWN method. As ground truth, we used four facial feature points, namely the position of two eyes, nose and mouth.

Figure 6 shows the correct feature localization rate for all test set. A feature was counted as accurately detected if it was localized to within 4 pixels of the ground-truth, in images of 128x192 pixels.



Feature	Detect Rate
Left Eye	96%
Right Eye	96%
Nose	91%
Mouth	92%

Figure 6: Feature localization results of our fully automatic system in more than 3000 FERET images. Face detection and face alignment take, in average, 100ms and 24ms per image, respectively.

Examples of correctly aligned face images are shown in Figure 7. Note that our method is quite robust to people with facial hair, wearing glasses, and exhibiting different facial expressions. In general, misalignments occurred due to significant illumination changes or when the position and scale of the face provided by the face detector falling outside the range supported by our algorithm. A multi-resolution approach could be used to enlarge the range and get even better results.

In this experiment, the average number of iterations for the AWN method was eight, thus requiring about 24ms to perform alignment. The face detector module requires about 100ms to process an image. Hence, the final system works in 124ms.

For the view-based approach, we collected a set of 68 sequences of 34 people, with two sequences per person. In each subrange, We selected 68 images for training. The speed and the localization accuracy is the same for the frontal view case.

We verified that our approach can be reliably applied for real-time facial feature tracking under large pose variation. Figure 8 shows some samples from a video sequence tracking demo.

As described in Section 3, the view model switching is based on a feature-based head pose estimator. In the first video frame, we assume the person is looking at the camera, so that we have a frontal view face, which can be automatically detected [16] and aligned using the frontal view range model. A generic 3D model is then constructed with a subset of the shape control points. In each view range, a set of nine visible non-coplanar points is used to estimate the pose, which selects the appropriate model for the next video frame.

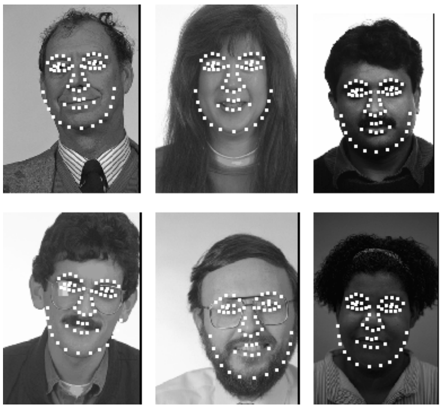


Figure 7: Examples of accurate face alignment in FERET database.

We assume that the face undergoes smooth motion, otherwise the wrong view model might be selected. We verified that the pose estimates are accurate enough to switch among the models. For more accurate pose computation, one should use a person-specific 3D model.

5. Conclusions

We have presented a view-based approach for automatic face alignment, based on a new method called Active Wavelet Networks (AWN). After optimizing the code, we obtained a significant enhancement in speed, making the method suitable for real-time applications. We also verified the robustness of the AWN algorithm with respect to unseen faces in a large dataset. Applications in alignment for face recognition and facial feature tracking under large pose variations were successfully demonstrated.

As future work, we plan to consider new view ranges to allow alignment and tracking also on pitch rotations. We also intend to use an explicit model for handling illumination changes.

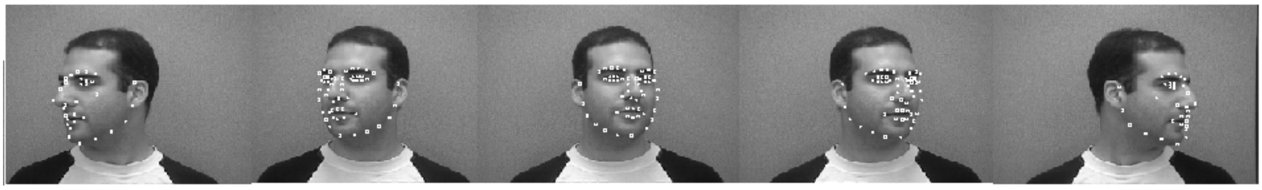


Figure 8: Samples frames showing real-time facial feature tracking under large pose variation.

Acknowledgements

This research was supported in part by NSR ITR grant #0205740.

References

[1] J. Ahlberg. Using the active appearance algorithm for face and facial feature tracking. In *ICCV'01 Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems*, Vancouver, BC, Canada, 2001.

[2] S. Baker and I. Matthews. Equivalency and efficiency of image alignment algorithms. In *Computer Vision and Pattern Recognition*, pages 1090–1097, 2001.

[3] T. Cootes, D. Cooper, C. Taylor, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[4] T. Cootes, G. Edwards, and C. Taylor. A comparative evaluation of active appearance model algorithms. In *British Machine Vision Conference*, pages 680–689, Southampton, UK, 1998.

[5] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001.

[6] T. Cootes, G. Wheeler, K. Walker, and C. Taylor. View-based active appearance models. *Image and Vision Computing*, 20:657–664, 2002.

[7] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, 1990.

[8] X. Hou, S. Li, H. Zhang, and Q. Cheng. Direct appearance models. In *Computer Vision and Pattern Recognition*, pages 828–833, 2001.

[9] C. Hu, R. Feris, and M. Turk. Active wavelet networks for face alignment. In *British Machine Vision Conference*, East Eaglia, Norwich, UK, 2003.

[10] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. In *First International Conference on Computer Vision*, pages 259–268, 1987.

[11] V. Krueger and G. Sommer. Gabor wavelet networks for object representation. *Journal of the Optical Society of America*, 2002.

[12] S. Li, X. Peng, H. Zhang, and Q. Cheng. Multi-view face pose estimation based on supervised isa learning. In *International Conference on Automatic Face and Gesture Recognition*, Washington, DC, USA, 2002.

[13] S. Li, S. Yan, H. Zhang, and Q. Cheng. Multi-view face alignment using direct appearance models. In *International Conference on Automatic Face and Gesture Recognition*, Washington, DC, USA, 2002.

[14] B.S. Manjunath and R. Chellappa. A unified approach to boundary perception: edges, textures, and illusory contours. *IEEE Transactions on Neural Networks*, 4(1):96–107, 1993.

[15] K. Toyama and G. Hager. Incremental focus of attention for robust vision-based tracking. *International Journal of Computer Vision*, 35(1):45–63, 1999.

[16] P. Viola and M. Jones. Robust real-time object detection. In *ICCV'01 Workshop on Statistical and Computation Theories of Vision*, Vancouver, BC, Canada, 2001.

[17] L. Wiskott, J. Fellous, N. Krueger, and C. Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:775–779, 1997.

[18] A. Yuille, D. Cohen, and P. Hallinan. Feature extraction from faces using deformable templates. *International Journal of Computer Vision*, 8(2):99–112, 1992.