Hierarchical Wavelet Networks for Facial Feature Localization

Rogério S. Feris Jim Gemmell Kentaro Toyama
Microsoft Research
Redmond, WA 98052
U.S.A.

Volker Krüger
University of Maryland, CFAR
College Park, MD 20742
U.S.A.

Abstract

We present a technique for facial feature localization using a two-level hierarchical wavelet network. The first level wavelet network is used for face matching, and yields an affine transformation used for a rough approximation of feature locations. Second level wavelet networks for each feature are then used to fine-tune the feature locations.

Construction of a training database containing hierarchical wavelet networks of many faces allows features to be detected in most faces. Experiments show that facial feature localization benefits significantly from the hierarchical approach. Results compare favorably with existing techniques for feature localization.

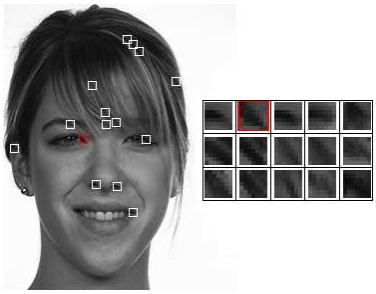

Figure 1. Candidates for an eye corner from a face image.

1. Introduction

Automated initialization of feature location is a requirement of many tracking algorithms that take advantage of temporal continuity of the target. In this paper, we describe an approach to automatic initialization using hierarchical wavelet networks. Our application is facial feature localization for the purpose of initializing facial feature tracking, but the approach is applicable to other target types.

Tracking algorithms that are based on tracking sets of compact visual features, such as edge corners or small image patches, are especially difficult to initialize because each feature in itself is rarely unique – brute-force raster-scan searches of such small features will result in many possible candidates, of which only a small handful may be desirable matches (Figure 1).

This suggests that features with larger support should be used, but features with larger support are also likely to be less precise in their localization, as image features far away from the feature in question bias localization. For example, many frontal face detectors [14, 15, 17] could trivially be converted to frontal eye detectors, by assuming that eyes are located at certain relative coordinates with respect to a detected face, and in fact, some face detectors overlay markers on the eyes, as evidence of a detected face [14, 15]. At a given resolution, whole faces contain more information than the eyes alone, and so the larger support of the face provides greater constraints in the search for eyes. On the other hand, the larger support also means that eye localization is imprecise because the face-eye relationship varies from image to image. Variations in facial geometry alone make it impossible to pinpoint pupils or eye corners using such a technique.

We present an algorithm which solves this problem via a hierarchical search using Gabor wavelet networks (GWNs, [8]). This approach allows effective object representation using a constellation of 2D Gabor wavelets that are specifically chosen to reflect the object properties.

For application to facial feature detection, we construct a training database of face images and their 2-level GWN representations. The first level GWN, representing the entire face, is used to find a face in the database that is similar to the target, and to determine an affine transformation to describe any difference in the orientation of the faces. The second level GWNs, representing each feature, are initialized in positions according to the affine transformation from the first level GWN. They are then allowed to move slightly to minimize their difference from the new face. This facilitates adjustments to account for slight differences in the geometry of the database face and the target. The final position of the child-wavelet networks is the estimate of the feature positions.

The remainder of the paper is organized as follows. In

Section 2, we explain Gabor wavelet networks, which form the basis for our approach, and introduce hierarchies of GWNs, as well. Section 3 discusses the algorithmic details of our feature-localization system and shows results on a hand-annotated database of faces and facial features. Finally, Section 4 reviews related work.

2. Wavelet Networks

A wavelet network consists of a set of wavelets and associated weights, where its geometrical configuration is defined with respect to a single coordinate system. It can be further transformed by a parametrized family of continuous geometric transformations. Wavelet Networks [20] have recently been adapted for image representation [8] and successfully applied to face tracking, recognition, and pose estimation [3, 8]. Here, we apply them to the problem of feature localization.

2.1 Basics

The constituents of a wavelet network are single wavelets and their associated coefficients. We will consider the odd-Gabor function as 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 [10]. Considering the 2D image case, each single odd Gabor wavelet can be expressed as follows:

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

where $\mathbf{x}$ represents image coordinates and $\mathbf{n}_i = (s^x, s^y, \theta, \mu^x, \mu^y)$ are parameters which compose the terms $\mathbf{S}_i = \begin{pmatrix} s_i^x \cos\theta_i & -s_i^y \sin\theta_i \\ s_i^x \sin\theta_i & s_i^y \cos\theta_i \end{pmatrix}$, and $\mu_i = \begin{pmatrix}\mu_i^x \\ \mu_i^y\end{pmatrix}$, that allow scaling, orientation, and translation. The parameters are defined with respect to a coordinate system that is held fixed for all wavelets that a single wavelet representation comprises. A Gabor wavelet network for a given image consists in a set of n such wavelets $\{\psi_{\mathbf{n}_i}\}$ and a set of associated weights $\{w_i\}$, specifically chosen so that the GWN representation:

$$\Psi(\mathbf{x}) = \sum_{i=1}^{n} w_i \psi_{\mathbf{n}_i}(\mathbf{x}) \qquad (2)$$

best approximates the target image.

2.2. Compression as Learning

Assuming we have a single training image, $\mathbf{I}^t$, that is truncated to the region that the target object occupies, we learn GWN representation parameters as follows:

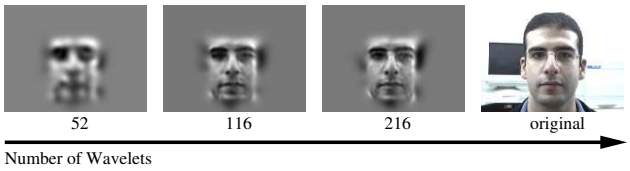

Figure 2. The image shows a facial reconstructions with variable accuracy, considering (from left to right) 52, 116 and 216 wavelets.

1. Randomly drop n wavelets of assorted position, scale, and orientation, within the bounds of the target object.
2. Perform gradient descent (e.g. via Levenberg-Marquardt optimization [12]) over the set of parameters $\{w_i, \mathbf{n}_i\}$, to minimize the difference between the GWN representation and the training image:

$$\arg\min_{w_i, \mathbf{n}_i} \left\| \mathbf{I}^t - \sum w_i \psi_{\mathbf{n}_i}(\mathbf{x}) \right\|^2. \qquad (3)$$

3. Save the geometric parameters, $\mathbf{n}_i$, and the weights, w_i, for all n wavelets. Let $\mathbf{v} = [w_1 w_2 \dots w_n]^T$ denote the concatenated vector of weights.

Step 2 minimizes the difference between the GWN representation of the training image and the training image itself. A reasonable choice of n results in a representation that is an effective encoding of the training image. One advantage of the GWN approach is that one can trade-off computational effort with representational accuracy, by increasing or decreasing n (see Fig. 2).

We note here that if the parameters for a wavelet, $\psi_{\mathbf{n}_i}(\mathbf{x})$, are fixed, then its coefficient, w_i, on an image, $\mathbf{I}$, can be computed easily from the image by taking the inner product of the wavelet's dual, $\tilde{\psi}_{\mathbf{n}_i}(\mathbf{x})$, with $\mathbf{I}$: $w_i = \langle \mathbf{I}, \tilde{\psi}_{\mathbf{n}_i} \rangle$. Here, $\langle \psi_{\mathbf{n}_i}(\mathbf{x}), \tilde{\psi}_{\mathbf{n}_j}(\mathbf{x}) \rangle = \delta_{i,j}$ (see [3, 8] for more details).

2.3. Localization

GWNs may be further transformed by a bijective geometric transformation, $\mathbf{T}_\alpha$, parametrized by α, such that the GWN representation $\Psi(\mathbf{x})$ is mapped to $\Psi(\mathbf{T}_\alpha^{-1}(\mathbf{x}))$. Localization of an object represented by Ψ can then be seen as finding the optimal parameters, α, of $\mathbf{T}$ that allow $\Psi(\mathbf{T}_\alpha^{-1}(\mathbf{x}))$ to best reconstruct a portion of the image. Given a hypothesized set of parameters, α, one way to determine whether it performs a good reconstruction is to compute $\Psi(\mathbf{T}_\alpha^{-1}(\mathbf{x}))$ and then compute the L_2-norm between it and the image (within Ψ's support region).

If the transformation $\mathbf{T}$ is linear it can be treated as being "pushed back" to the individual wavelets, $\psi_{\mathbf{n}_i}(\mathbf{x})$, that make up the GWN representation. In this case, we do not have to laboriously reconstruct images to compute the L_2-norm.

Instead, given a hypothesized set of parameters, α, we can now transform the constituent wavelets accordingly, compute their weights, w, on the image, $\mathbf{I}$, and directly compute L_2-norm as follows:

$$\begin{aligned}\|\mathbf{I} - \Psi(\mathbf{T}_\alpha^{-1}(\mathbf{x}))\|^2 &= \|\mathbf{v} - \mathbf{w}\|_\Psi^2 \\ &= \sum_{i,j}(v_i - w_i)(v_j - w_j)\langle\psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_j}\rangle \ , \end{aligned} \quad (4)$$

where $v_i = \langle \mathbf{I}(\mathbf{x}), \tilde{\psi}_{\mathbf{n}_i}(\mathbf{T}_\alpha^{-1}(\mathbf{x}))\rangle$.

The terms $\langle\psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_j}\rangle$ are independent of α up to a scalar factor, thus further facilitating on-line computations.

2.4. Hierarchical Wavelet Networks

Hierarchical wavelet networks are best envisioned as a tree of wavelet networks. Each node of the tree represents a single wavelet network together with its coordinate system. Each child node is associated with a fixed local coordinate system that is positioned, scaled, and oriented with respect to its parent. Child nodes represent wavelet networks in themselves. Relationships between the wavelet parameters in a parent node and a child node are not fixed *a priori*. That is, this hierarchical structure only imposes direct constraints on the relative positioning of coordinate systems between nodes, not on the wavelets themselves.

Structured in this way, wavelet networks occurring higher (toward the root) in the tree constrain their child-node wavelet networks in such a way as to avoid significant geometric deviations while offering enough flexibility that local distortions can still be modeled.

3. Implementation

Our test system was developed to provide initialization for a 3D facial pose tracker. The tracking system (described in [4, 16]) uses nine tracked features on a subject's face – inner and outer corners of both eyes, three points on the nose, and two mouth corners. Each feature is tracked by a combination of low-resolution, sum-of-absolute-differences template matching and iterative sub-pixel tracking of small image patches [6, 9]. Both feature-tracking algorithms require accurate initial localization of the nine features, per subject, in order to track. Previously, these points were initialized manually for each subject; by implementing the algorithms described above, we were able to automate this process for a range of subjects. In the remaining sequences, *facial features* will refer to eight of these features (not including the nose tip – this is estimated as the midpoint between nostrils, because local image information is insufficient for accurate localization).

3.1. Training Database

Our training database includes the following for each face:

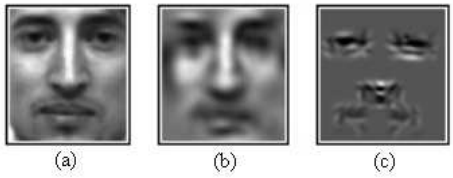

Figure 3. Training database: (a) face image (b) GWN representation of face (c) GWN of features.

- the original image,
- a bounding box for each facial feature,
- a bounding box for the whole face,
- a GWN representation of the region inside the face bounding box, and
- a GWN representation of the region inside each facial feature bounding box.

Faces are well-represented with a GWN of 52 wavelets, as shown in Figure 2 (Cf. the Gabor jet approach, which would require many more wavelets). Each facial feature is represented by a GWN comprising nine wavelets.

3.2. Level One: Face Matching

Assume we are given an image known to have a face present together with the approximate location of the face (*e.g.*, via face detection [14, 15, 17]). The first step in feature localization we call *face matching*. The task is to find the "best match" face from our database of faces, using the first level of the GWN hierarchy and a nearest-neighbor algorithm.

For each candidate face, we begin by determining an affine transformation of the level-one GWN that registers the candidate with the target image, as explained in Section 2.3. Levenberg-Marquardt optimization was used to find the best affine parameters. The residual score in wavelet subspace (Equation 4) is then minimized over candidates to suggest the best-match face from our database. Intuitively, this score gives an indication of how good a candidate is, for the purposes of initialization of Level Two, below.

Note that at this point, we can generate reasonable hypotheses for feature positions already, simply by applying the affine transformation to the relative positions of the features with respect to the whole face, as marked in our database. The success rate of these first-level hypotheses is given in Table 1.

In the next subsection, we show how these estimates are further refined by level-two analysis.

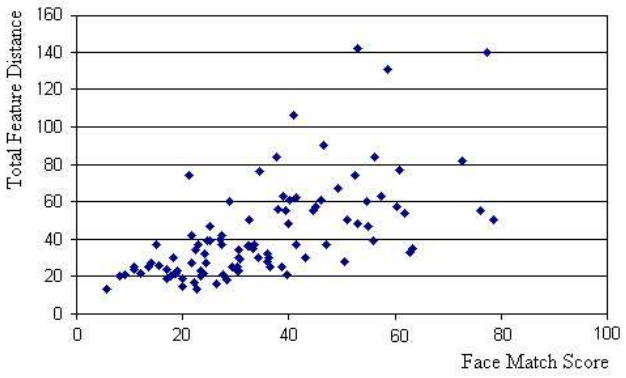

Figure 4. First-level matching: Sum of feature position differences vs face match score for one face.

3.3. Level Two: Feature Localization

Level One gives us an initial starting point for finer search. The refinement process is identical in the abstract to how we computed the affine transformation in Level One. The details are slightly different:

We do not allow arbitrary affine transformations for facial features, because local features tend to have far fewer image constraints. A problem akin to the "aperture effect" comes into play, and this is aggravated by searching over too many degrees of freedom. Since we already know the facial orientation, scaling and expected aspect ratio from Level One, we restrict our search to translational parameters, only.

For each feature, we perform a brute-force search within a limited window for a position that minimizes the score in wavelet subspace between a candidate level-two feature GWN, and the target image.

Note that candidate feature GWNs may be drawn from *any* of the faces in our database, not just the GWNs that are associated with the best-match face from Level One. This gives even a relatively small database the power to match a considerable segment of the population, by mixing and matching features from different faces.

3.4. Results

Experimental validation of our approach was obtained by constructing a database of 100 faces, drawn from the Yale and FERET Face Databases [1, 11]. To test, we performed a leave-one-out series of 100 experiments, where for each face, we apply feature localization using the remaining database of 99 faces. For each set of automated feature localizations, we compare with the hand-marked locations of each feature.

Feature	1-level detect rate	2-level detect rate
Left Eye Outside Corner	0.81	0.95
Left Eye Inside Corner	0.90	0.94
Right Eye Inside Corner	0.93	0.94
Right Eye Outside Corner	0.78	0.96
Left Nostril	0.86	0.95
Right Nostril	0.88	0.94
Left Lipcorner	0.65	0.87
Right Lipcorner	0.65	0.88

Table 1. Table of feature localization accuracy for 1- and 2-level hierarchies. A feature was counted as accurately detected if it was localized to within 3 pixels of the point marked by hand.

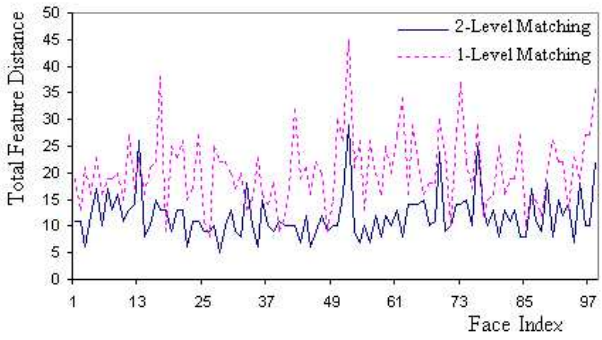

Figure 5. Sum of feature position differences, for each face, for 1- and 2-level systems.

Figure 4 plots the sum of feature position differences versus face score for a single face, with all other faces in the database scored against it. This figure demonstrates that a good score always corresponds to a small position difference. To show that there is considerable advantage to additional layers in the hierarchy, we compare feature localization results using only one level to using both levels. Table 1 compares feature localization rates for both 1- and 2-level systems. An "accurate" localization is characterized as one in which the feature was localized to within 3 pixels (L_2-distance) of the hand-marked position. Note that features are localized consistenly more accurately for all features with two levels rather than one. Figure 5 shows this same trend broken down differently. The solid line indicates the total SAD in feature position between 2-level localization and hand-annotation; the dashed line is for 1-level localization. Except in a two or three rare instances, the 2-level localization is far superior.

Finally, we offer random examples out of the 100 experiments for visual examination. Figure 6 shows a clear improvement in feature localization with two levels. Note that

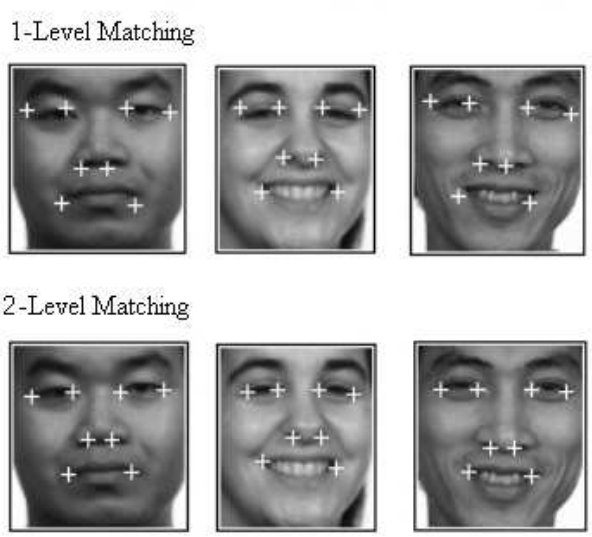

Figure 6. Feature detection results. Improved accuracy by using hierarchical localization.

just about every feature is accurately localized by two-level matching.

Figures 7 and 8 illustrate further cases of accurate and inaccurate detection cases using the two-level hierarchy. Figure 8 shows examples of some rare failure cases. Among failures, these examples are typical – eyebrows or shadows under the eyes are sometimes mistaken for the eyes themselves, and specular reflection from glasses can obfuscate eye corners.

4. Related Work

Other facial feature detection approaches exist. One approach detects feature points using hand-crafted geometric models of features [19]. The goal of this work, however, is in detection of faces by looking for groups of facial features, so feature localization accuracy is low. Other work trains separate face and facial feature detectors, where features are trained for maximum discriminability from among a training set [2]. This work is presented without quantitative measures of feature localization. Steerable filters and geometrical models have also been used to find facial features with high accuracy[7]. A coarse-to-fine image pyramid is employed to localize the features, but the technique requires high-resolution imagery in which sub-features such as the whites of the eye are clearly visible as such. Color segmentation can also be used to estimate approximate feature locations [5]. These estimates, reported to have a precision of up to ± 2 pixels, can be further refined via grayscale templates to sub-pixel accuracy. For each individual and each face feature nine 20×20 pixel templates are given, but no generalization to unknown faces is discussed. Finally, neural networks have been used to detect eyes and eye corners [13]. Results approach 96% correctly detected eye corners while allowing a variance of two pixels, but these results are for eyes only, which are less deformable than mouths.

Lastly, GWNs invite the closest comparison with the well-known Gabor jet representations of facial features [18]. The advantage of GWNs is that they offer a sparser representation of image data: Where jets can require up to 40 complex Gabor filters to approximate the local image structure around a single feature point, GWNs can make do with nine, as in our implementation. This is a direct consequence of allowing wavelets in a GWN to roam continuously in their parameter space during training. Edge features, which are building blocks of more complex features, are thus efficiently captured at various scales by GWNs.

5 Conclusion

We have presented a hierarchical wavelet network approach to feature detection. Our method takes a coarse-to-fine approach to localize small features, using cascading sets of GWN features.

We tested our results on the task of facial feature localization, using one- and two-level hierarchies. For the one-level implementation, GWNs are trained for the whole face; for two levels, the second-level GWNs are trained for each of eight facial features. Experiments show that the two-level system outperforms the one-level system easily, verifying the usefulness of a hierarchy of GWNs for feature localization. Results compare favorably with other algorithms on this task.

Some remaining issues include the following: How can we determine the minimum number of wavelets required for each GWN? Can a subset of wavelets in a given network be sufficient for good matching at a particular level? Finally, how can we minimize the number of GWNs necessary at each level to capture the broad range of the set of real targets? We hope to examine these questions as future work.

References

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Patt. Anal. and Mach. Intel.*, 19(7):711–720, 1997. Special Issue on Face Recognition.

[2] A. Colmenarez, B. Frey, and T. Huang. Detection and tracking of faces and facial features. 1999.

[3] R. Feris, V. Krüger, and R. C. Jr. Efficient real-time face tracking in wavelet subspace. In *Proceedings of the Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems*, Vancouver, BC, Canada, 2001, in conjunction wit the ICCV'01, 2001.

[4] J. Gemmell, K. Toyama, C. L. Zitnick, T. Kang, and S. Seitz. Gaze awareness for video-conferencing: a software approach. *IEEE Multimedia*, 7(4), October 2000.

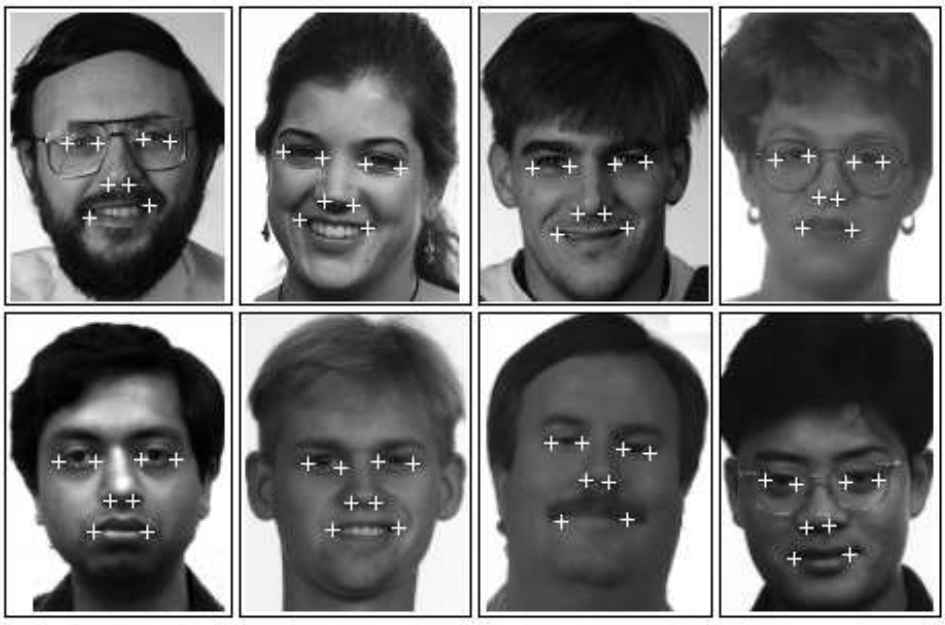

Figure 7. Feature detection results. Examples of accurate detection.

[5] H. Graf, E. Casotto, and T. Ezzat. Face analysis for synthesis of photo-realistic talking heads. In *Proc. Int'l Conf. on Autom. Face and Gesture Recog.*, pages 189–194, Grenoble, France, March, 28-30, 2000.

[6] G. Hager and P. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *PAMI*, 20(10):1025–1039, October 1998.

[7] R. Herpers and et al. Edge and keypoint detection in facial regions. In *Killington, VT, Oct. 14-16*, pages 212–217, 1996.

[8] V. Krüger. Gabor wavelet networks for object representation. Technical Report CS-TR-4245, University of Maryland, CFAR, May 2001.

[9] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. Int'l Joint Conf. on AI*, pages 674–679, 1981.

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

[11] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The feret evaluation. In H. W. et al., editor, *Face Recognition: From Theory to Applications*, pages 244–261, 1998.

[12] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes, The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 1986.

[13] M. Reinders, R. Koch, and J. Gerbrands. Locating facial features in image sequences using neural networks. 1997.

[14] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. Patt. Anal. and Mach. Intel.*, 20:23–38, 1998.

[15] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *Proc. Computer Vision and Patt. Recog.*, pages 749–751, Hilton Head Island, SC, June 13-15, 2000.

[16] K. Toyama and G. Hager. Incremental focus of attention for robust vision-based tracking. *Int'l J. of Computer Vision*, 35(1):45–63, 1999.

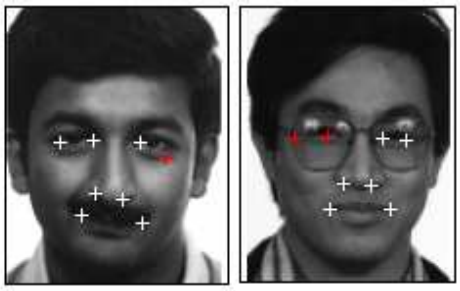

Figure 8. Feature detection results. Examples of inaccurate detection.

[17] P. Viola and M. Jones. Robust real-time face detection. In *ICCV01*, page II: 747, 2001.

[18] L. Wiskott, J. M. Fellous, N. Krüger, and C. v. d. Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. Patt. Anal. and Mach. Intel.*, 19:775–779, 1997.

[19] K. Yow and R. Cipolla. Feature based human face dection. *Image and Vision Computing*, 15:713–735, 1997.

[20] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Trans. Neural Networks*, 3:889–898, 1992.