Detection and Tracking of Faces and Facial Features

Antonio Colmenarez
Adaptive Systems Department
Philips Research
Briarcliff Manor, NY 10510
USA

Brendan Frey
Department of Computer Science
University of Waterloo
Waterloo, Ontario N2L 3G1
Canada

Thomas S. Huang
Beckman Institute
University of Illinois
Urbana, IL 61801
USA

Abstract

We describe a real-time system for face and facial feature detection and tracking in continuous video. The core of this system consists of a set of novel facial feature detectors based on our previously proposed Information-Based Maximum Discrimination learning technique. These classifiers are very fast and allow us to implement a fully automatic, real-time system for detection and tracking multiple faces. In addition to locking onto up to four target faces, this system locates and tracks nine facial features as they move under facial expression changes.

1 Introduction

In this paper, we present in detail a fully automatic, person-independent, real-time system for detection and tracking multiple faces and nine facial features. We use Information-Based Maximum Discrimination classifiers [1, 2] with a novel set of low-level image features to locate accurately and efficiently nine facial features including non-rigid points as they move under facial expressions.

2 Tracking and Motion Analysis

Visual object tracking and motion analysis from video are areas of great importance in computer vision. The main objective of tracking is to roughly predict and estimate the location of the target object in each frame of the image sequence despite changes in the object's pose, size, illumination and appearance. Motion analysis is concerned with the estimation of the non-rigid motion within the parts of the object being tracked.

Figure 1 illustrates a general scheme for object tracking and motion analysis. Note that we have highlighted two different loops: (i) the tracking loop, which executes at the frame rate of the input video, and (ii) the initialization loop, which executes only at the beginning or when the confidence level of the tracking loop drops below some acceptable level.

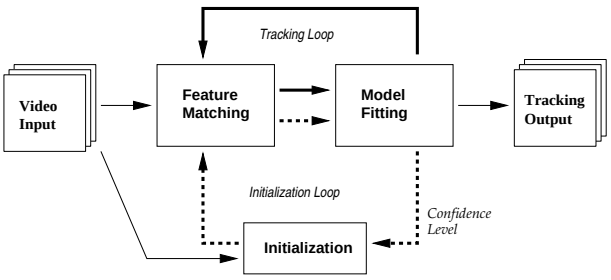

Figure 1: General Scheme for Visual Tracking and Motion Analysis.

The tracking loop is further divided in two steps: the feature matching, and model fitting. The feature matching step is responsible for locating object features using image processing and pattern recognition algorithms, while the model fitting step imposes geometrical constraints and combines the individual results of the feature locations based on the knowledge provided by the object model.

In most approaches, the initialization is manual. The locations of the features are specified by hand or the user positions the features in preset locations. Then, the geometry of the model as well as the data used in the feature matching step are adjusted to perform the tracking in the subsequent frames.

2.1 Previous Approaches

In early tracking systems [3, 4, 5], the feature matching step was carried out from one frame to the next using optical flow computations, resulting in drifting errors accumulating over long video sequences. In later techniques, feature texture information is gathered during initialization, so the feature matching step is carried out with respect to the initialization frame to overcome drifting.

In order to deal with large out of the plane rota-

tions, a 3D model of the geometry of the face has been used together with the texture obtained from the initialization step to achieve 3D pose estimation simultaneously with face tracking in an analysis-by-synthesis scheme [6, 7, 8]. In this approach, the 3D model is used to create the templates by rendering the texture given the head pose so that the feature matching step performs well on large out of the plane rotations. However, this system requires the 3D model of the person's head/face.

A wire-frame model capable of non-rigid motion has also been used to analyze facial expressions together with the global position of the face [9]. In a more complex scheme [10], optical flow constrains are used together with a wire-frame model to track rigid and non-rigid motion and adapt the wire-frame model to fit the person's head. One of the most serious limitations in the wire-frame approaches is the fitting of the wire-frame model to the face in the initialization frame; this task involve the accurate location of many facial feature points and is carried out by hand.

Other approaches of interest are those based on "blobs", where, the face and other body parts are modeled with 2D- or 3D-Gaussian distributions of of pixels. Pixels are clustered by their intensity [11], color [12], or even disparity maps from stereo images [13]. Although these techniques fail to capture non-rigid facial motion, they are easily initialized and operate very efficiently especially even in sequences with moderate occlusion.

In general, algorithms that use complex wire-frame models provide a framework for high level motion analysis of non-rigid facial motion. However, these complex models need to be customized to the face being tracked during a similarly complex initialization procedure. At the other end of the spectrum, algorithms based on simple models, such as blobs, have proven to be feasible. Their simple initialization procedures and low computational requirements allow them to run in real-time on portable computers, but they are limited in the amount of information they extract from the object parts.

3 Face and Facial Feature Tracking

In this section, we describe in detail a face and facial feature tracking system based on Information-Based Maximum Discrimination (IBMD) classifiers. These facial feature classifiers are used not only to initialize the tracking system, but also as the matching kernels in the main tracking loop. Tracking is invariant to rotation and scale so that once the tracker locks onto the face, it is allowed to get close or far from the camera and to rotate freely.

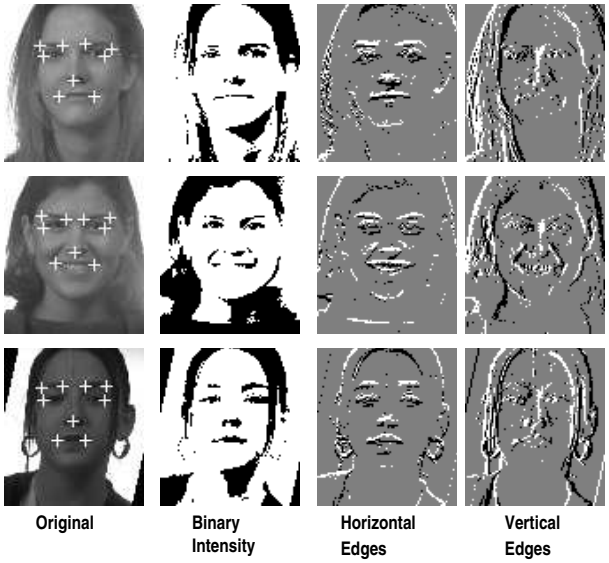

Figure 2: Examples of Training Images and Low-Level Image Features

The input of the system is continuous video which may contain multiple faces. In addition to providing a log of the time when people step in and out of the field of vision and the location of the faces and nine facial features as they are being tracked, the system performs temporal analysis of the global position of the faces to detect head shaking and nodding.

3.1 Information-Based Learning

Feature matching is carried out in a maximum likelihood setup with facial feature classifiers trained with facial feature examples using Information-Based Maximum Discrimination Learning.

Consider a two-class discrimination problem where $\mathbf{y} \in \{0, 1\}$ is the class index and $\mathbf{x}$ is a discrete observation vector from which we are to guess the class using a classifier $\mathbf{y} = g(x)$. We use a classifier based on the likelihood ratio,

$$\mathbf{L}(\mathbf{x}, \mathbf{s}) = \frac{\mathbf{P}(\mathbf{x}|\mathbf{s}, \mathbf{y} = 1)}{\mathbf{P}(\mathbf{x}|\mathbf{s}, \mathbf{y} = 0)}, \tag{1}$$

where $\mathbf{s}$ are some parameters to be learned from the training examples. Our learning algorithm maximizes the information-theoretic divergence (Kullback-Leibler divergence),

$$\mathbf{H}(\mathbf{s}) = \sum_{\mathbf{x}} \mathbf{P}(\mathbf{x}|\mathbf{s}, \mathbf{y} = 1) \log \frac{\mathbf{P}(\mathbf{x}|\mathbf{s}, \mathbf{y} = 1)}{\mathbf{P}(\mathbf{x}|\mathbf{s}, \mathbf{y} = 0)}, \tag{2}$$

which is a quantity that measures the discrimination capability of the classifier.

A modified, 1st order Markov model is used to model the pixel probabilities and the order of the pixels is specified by $\mathbf{s}$:

$$\mathbf{P}(\mathbf{x}|\mathbf{s},\mathbf{y}) = \mathbf{P}(x_{s_1}|\mathbf{y}) \prod_{m=2}^{d} \mathbf{P}(x_{s_m}|x_{s_{m-1}}|\mathbf{y}) \quad (3)$$

So, s_i is the index of the pixel at the i-th stage of the Markov model. The likelihood in (1) is computed from

$$\mathbf{L}(\mathbf{x},\mathbf{s}) = \mathbf{L}(x_{s_1}) \prod_{m=2}^{d} \mathbf{L}(x_{s_m}|x_{s_{m-1}}) \quad (4)$$

and the divergence in (2) from

$$\mathbf{H}(\mathbf{s}) = \mathbf{H}(s_1) + \sum_{m=2}^{d} \mathbf{H}(s_m, s_{m-1}) \quad (5)$$

Learning proceeds by computing independent divergences $\mathbf{H}(i)$ and pair-wise divergences $\mathbf{H}(i,j) \quad i,j = 1,\ldots,d$ and then solving the optimization $\mathbf{s}^* = \arg\max \mathbf{H}(\mathbf{s})$ using a greedy algorithm. Once we have found a good, suboptimal solution $\mathbf{s}'$, we use the logarithm of the likelihood in (4):

$$\log \mathbf{L}(\mathbf{x},\mathbf{s}') = \log \mathbf{L}(x_{s'_1}) + \sum_{m=2}^{d} \log \mathbf{L}(x_{s'_m}|x_{s'_{m-1}}) \quad (6)$$

as the discrimination function. Note that an observation vector containing d pixels from an image window is classified with only d additions.

3.2 Facial Feature Matching

We train the facial feature classifiers using positive examples of the features from the FERET database [14, 15]. Negative examples are obtained from subwindows near the locations of the facial features.

The discrete observation vector or image subwindow of each facial feature consists of the combination of three distinct low-level image features: binary intensity, vertical edges (3 levels) and horizontal edges (3 levels), so that each element of this discrete observation vector has $3 \times 3 \times 2 = 18$ possible values. Examples of the training images and of these low-level image features are shown in Figure 2.

Since the classifiers are trained to perform within a limited range of scale and rotation , a scheme for rotation and scale invariant matching/tracking was developed. As illustrated in Figure 3, the predicted position, orientation and size of the face is used to project a region of the input frame onto an image where matching is performed. Once the features are found in this image, they are projected back to the input frame.

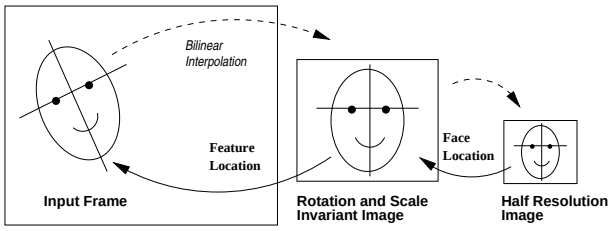

Figure 3: Hierarchical, Scale- and Rotation-Invariant Feature Location Scheme.

In order to deal with fast motion from one frame to another, a hierarchical matching procedure was implemented. At half resolution, a classifier first detects the whole face in a search area of size similar to that of the face. Then, the rest of the features are located at full resolution using much smaller search areas in appropriate positions relative to the face.

3.3 Model Fitting

In this tracking system, the IBMD classifiers act as person-independent models for the local appearances of the facial features. The geometrical constrains of the relative position of the features are obtained from the training examples and consist of the location and size of the search areas of the feature detectors in the scale and rotation invariant image. Consequently, the feature location accuracy of this system is the same as that of the feature classifiers. The confidence level for the tracking procedure is a combination of the confidence levels for the feature detectors.

3.4 Initialization

The described tracking system deals with multiple faces and perform in real-time. Figure 4 shows a diagram of how the system operates. The main tracking loop, indicated as "Real-Time Tracking", executes synchronously with the video frame grabber, processing one frame per period. Initialization is needed much less frequently and consists of two steps: "Object Detection" and "Behind Tracking," the tracking of the frames left behind.

Our face detection algorithm searches for faces using frontal views with up-right orientations (less than ± 6.0 degree rotation) and size ranging from 25 to 407 pixels. It locates face candidates and their outer eye corners.

Assume that at time $t = 0$, the initialization procedure is started with an execution of the face detection algorithm. By time t_1 when the face detection is completed, a total of $N = t_1/T_{video}$ frames have passed and the face might have moved too far from its original position at frame $t = 0$ for tracking to work.

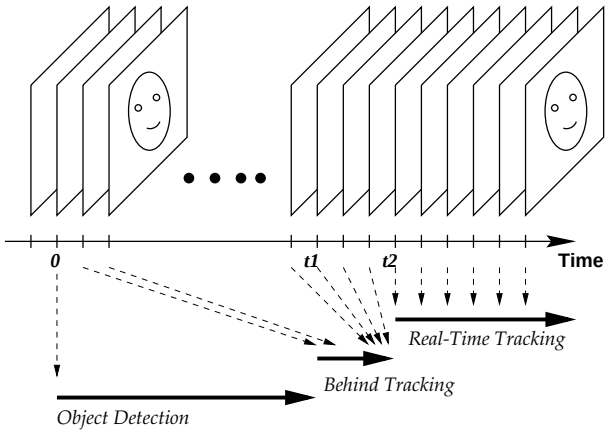

Figure 4: Initialization in a Real-Time Tracking System.

During the time period indicated as "Behind Tracking", the tracking algorithm executes asynchronously and as fast as possible, starting from the frame immediately after $t = 0$ until it catches up with the video frame grabber. Then, real-time operation resumes.

In a system intended to deal with multiple faces, the initialization procedure should execute continuously to guarantee that faces entering the vision field are locked onto. To reduce the computational requirements in our implementation, we force the initialization procedure to execute only once every T_{init} seconds, and to stop once the maximum number of tracked faces N_{faces} is reached. Default values, $T_{init} = 1$ seconds and $N_{faces} = 4$, produce a total latency of less that 2 seconds and leave plenty of processing power available in the computer for other tasks.

4 Results

We have quantitatively evaluated the performance of our tracking system using a database of video segments of head-and-shoulder scenes intended for different aspects of facial analysis. This database consist of 54 video segments containing 18 people with 3 distinct video segment each. We stored the videos in MPEG1 format, 320×240 color pixels, at 30 frames per seconds at an approximate rate of 1 Mbits/sec.

Out of the 107460 image frames in this video database, the face was successfully tracked on 105417 (98%). The nine facial features detected are the outer eye corners, the four eyebrow corners, the center of the nostrils and the mouth corners. Figure 5 shows several sample frames from our video database and the result of our tracking system.

Figure 5: Sample frames from our Face Video Database and the results of the facial feature detection/tracking system.

4.1 Performance of the Facial Feature Classifier

We have also perform a detailed evaluation and comparison of the performance of our facial feature classifier under "error boostrapping" [1] and other parameter adjustments in the learning technique. We have measured the detection accuracy and the ROC for each of the feature detectors. Space limitations in this paper prevent us from show these results in detail. Interested reader are referred to **??**.

From the performance comparisons of these facial feature locators, two important observations were made. First, the center of the nostrils are by far the easiest facial feature to detect followed by the eye corners. Second, there is a trade off between accuracy and robustness with respect to the size of the facial feature windows. Larger feature windows resulted in more robust feature detection and less accuracy in the feature location.

5 Concluding Remarks

We have presented a novel algorithm for detecting facial features and an implementation of a real-time system for tracking multiple faces and facial features. The facial feature tracking results on our video

database are used to conduct tests of face recognition and facial expression recognition algorithms [16, 17].

Our face detection and tracking system is part of a multimedia, multi-modal, web-based Computer Travel Companion; interested readers are referred to http://empath.vp.uiuc.edu/TravelCompanion/ for more details. In addition to detection and tracking, the system analyzes the absolute position of tracked faces. The algorithm detects shaking and nodding of the head, different viewing scenarios (such as near/distant, upright, lying down) and periods of high & low activity. This analysis is useful to sense not only explicit events such as yes- and no-like gestures, but many others important clues about the user's state of mind. This information is used for user customization and adaptation in advanced human-computer interfaces.

Acknowledgments

This work was supported by Army Research Laboratory under Cooperative Agreement No. DAAL01-96-2-0003. Brendan Frey was a Beckman Fellow at the time this research was conducted.

References

[1] A. J. Colmenarez and T. S. Huang, "Pattern detection with information-based maximum discrimination and error bootstrapping," in *Proc. ICPR*, 1998.

[2] A. J. Colmenarez and T. S. Huang, "Face detection with information-based maximum discrimination," in *Proc. CVPR*, 1997.

[3] H. Li, P. Roivainen, and R. Forchheimer, "3-d motion estimation in model-based facial image coding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 545–555, June 1993.

[4] S. Basu, I. Essa, and A. Pentland, "Motion regularization for model-based head tracking," MIT Media Laboratory Perceptual Computing Section, Cambridge, MA, Tech. Rep. 362, Aug. 1996.

[5] I. Essa, T. Darnell, and A. Pentland, "Tracking facial motion," in *Proceedings of IEEE Nonrigid and Articulated Motion Workshop*, (Austin, TX), November 1994, pp. 36–42.

[6] A. Colmenarez, R. Lopez, and T. Huang, "3d model-based head tracking," in *Visual Communications and Image Processing*, (San Jose, CA USA), SPIE, SPIE Press, Feb. 1997, p. TBA.

[7] R. Lopez, A. Colmenarez, and T. Huang, "Vision-based head and facial feature tracking," in *Displays Fed Lab Annual Symposium*, (Adelphi, MD), Jan. 1997.

[8] T. S. Huang, R. Lopez, H. Tao, and A. Colmenarez, *Visual Communication and Image Processing*, ch. 3D Model-based Image Communication. Optical Engineering Series, Marcel Dekker, Inc., 1997.

[9] H. Tao and T. S. Huang, "Connected vibrations: a modal analysis approach to non-rigid motion tracking," in *International Conference on Computer Vision and Pattern Recognition*, 1998.

[10] G. Bozdagi, M. Tekalp, and L. Onural, "3-d motion estimation and wireframe adaptation including photometrics for model-based coding of facial image sequences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, pp. 246–256, June 1994.

[11] A. Ali and A. Pentland, "Real-time self-calibrating stereo person tracking using 3-d shape estimation from blob features," in *13th International Conference on Pattern Recognition*, 1996.

[12] R. Quian, I. Sezan, and K. Matthews, "A robust real-time face tracking algorithm," in *International Conference on Image Processing*, 1998.

[13] N. Jojic, M. Turk, and T. S. Huang, "Tracking articulated structures in dense disparity maps," in *Submitted to CVPR*, 1999.

[14] P. J. Phillips, H. Moon, S. Rizvi, and P. Rauss, "The feret evaluation," in *Proc. NATO-ASI on Face Recognition: From Theory to Applications*, 1997, pp. 244–261.

[15] P. J. Phillips, H. Moon, S. Rizvi, and P. Rauss, "The feret evaluation methodology for face-recognition algorithms," in *CVPR*, 1997, pp. 137–143.

[16] B. Frey, A. Colmenarez, and T. Huang, "Mixtures of local linear subspaces for face recognition," in *CVPR*, 1998, pp. 32–37.

[17] A. Colmenarez, B. Frey, and T. Huang, "A probabilistic framework for embedded face and facial expression recognition," in *Submitted to CVPR*, 1999.