Segmentation and Tracking of Faces in Color Images

Karin Sobottka Ioannis Pitas
Department of Informatics
University of Thessaloniki 540 06, Greece
E-mail: {sobottka, pitas}@zeus.csd.auth.gr

Abstract

In this paper we present a new approach for automatically segmentation and tracking of faces in color images. Segmentation of faces is performed by evaluating color and shape information. First, skin-like regions are determined based on the color attributes hue and saturation. Then regions with elliptical shape are selected as face hypotheses. They are verified by searching for facial features in their interior. After a face is reliably detected, it is tracked over time. Tracking is realized by using an active contour model. The exterior forces of the snake are defined based on color features. They push or pull snaxels perpendicular to the snake. Results for tracking are shown for an image sequence consisting of 150 frames.

1. Introduction

Due to variations in illumination, background, visual angle and facial expressions, machine based face recognition is a complex problem. Although a lot of work has already been done in this research field ([2], [3]), robust face detection, tracking or recognition is still a problem. The present paper deals with the topic of face segmentation and tracking. As it is described in [1], this approach is integrated in a multi-modal system for person verification using speech and image information. Face segmentation and tracking can be done based on edges [6] or, as we shall show here, based on regions. The advantage of considering regions is that they are more robust against noise and changes in illumination. First we segment faces by using color and shape information. Then facial features are extracted to verify the face hypotheses. If a face is reliably detected, its contour is tracked over time by using an active contour model.

2. Face detection

We perform face segmentation by thresholding the color input image using predefined domains of hue and saturation that describe the human skin color. Afterwards we check for each region with skin color if its shape is elliptical or not. Face candidates are verified by searching for facial features in their interior.

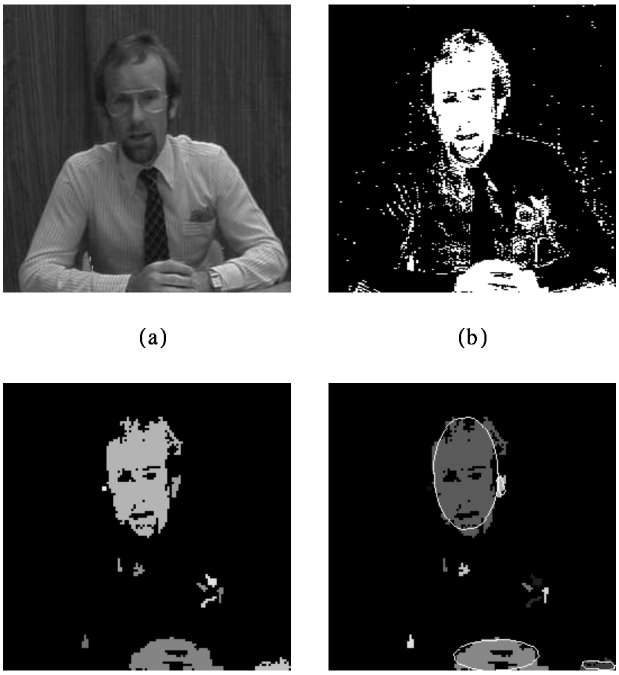
2.1. Face segmentation

The segmentation of faces out of complex scenes can be done robustly on the basis of color and shape information ([11], [12]). The effectiveness of using color information was also shown in [5]. Mostly the primary components of Red, Green and Blue are used for skin segmentation [13]. In our case we consider the Hue-Saturation-Value (HSV) color space, because it is very similar to the human perception of colors. Alternatively, similar color spaces can be used as well (e.g. HSI, HLS). For the segmentation of skin-like regions it is sufficient to use appropriately defined domains of hue and saturation. By disregarding the value (luminance) component, robustness is obtained towards changes in illumination and shadows. An example for such a segmentation is shown in Figure 1a,b.

Because the human face is more or less a connected region with skin color, we perform connected component analysis on the segmented image. Results of this step are illustrated in Figure 1c. Afterwards for each connected component shape information is evaluated. The oval shape of faces can be approximated by ellipses. Therefore, we compute the best-fit ellipse E for each connected component C on the basis of moments [12]. Then we assess, how well C is approximated by its best-fit ellipse. For that purpose the following measure V is evaluated:

$$V = \frac{\sum_{(x,y)\in E}(1-b(x,y)) + \sum_{(x,y)\in C\setminus E} b(x,y)}{\sum_{(x,y)\in E} 1} \quad (1)$$

where $b(x,y)$ denotes the indicator function of C. V determines the distance between the connected component and the best-fit ellipse by counting the "holes" inside of the ellipse and the points of the connected component which are outside of the ellipse. Connected components which are well approximated by their best-fit ellipse are considered as



(a) (b)

(c) (d)

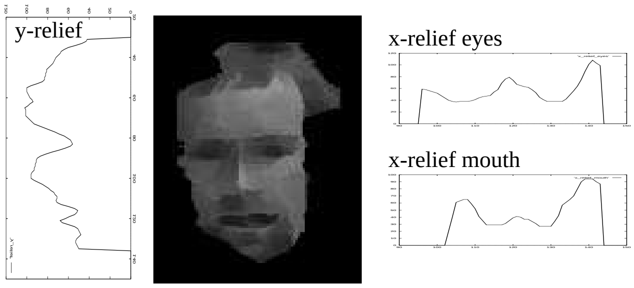
Figure 1. Face segmentation: (a) original color image, (b) skin segmentation, (c) connected components and (d) best-fit ellipses.

face candidates (Figure 1d). They are verified by searching for facial features inside of the connected components.

2.2. Facial feature extraction

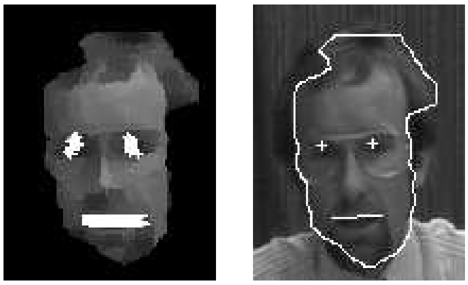
Our approach for facial feature extraction is based on the observation that, in intensity images, facial features differ from the rest of the face because of their low brightness. This observation is also used in [10]. In case of the eyes, reasons for that are the color of the pupils and the sunken eye sockets. Thus, in the following, we consider intensity information in the interior of the connected components. In a preprocessing step, we enhance facial features by applying greyscale erosion and an extremum sharpening operation. Results of this preprocessing step are shown in Figure 2b.

Afterwards facial features are extracted by the analysis of minima and maxima. For that we evaluate the projections of the topographic greylevel relief of the connected component. First the y-projection is determined by computing the mean greylevel value of every row of the connected component (Figure 2a). Then minima and maxima are searched in the smoothed y-relief. By checking the gradients, significant minima are selected. For each significant minimum of the y-relief, x-reliefs are computed by averaging the greylevel values of 3 neighboured rows of every column (Figure 2c). After smoothing the x-reliefs, minima and maxima are determined. Beginning with the uppermost minima of the y-relief, we search through the lists of minima and maxima of x-reliefs to find facial feature candidates. For example in case of the eyes, we search for two minima that meet the requirement for eyes concerning relative position inside of head, significance of maximum between minima, ratio of distance between minima to head width and similarity of greylevel values. In case of the mouth we look for two maxima that form the borders of the mouth region. As result we obtain a set of facial feature candidates. Examples of eye and mouth candidates, represented by crosses and line segments, are shown in Figure 3a.



(a) (b) (c)

Figure 2. Min-max analysis



(a) (b)

Figure 3. (a) Eyes and mouth candidates and (b) facial features and face contour.

After candidates are determined, we cluster them according to their left and right x-coordinates. By this step, we reduce the number of candidates significantly and obtain representative candidates for facial features. Then we build all possible face constellations and assess each of them based on vertical symmetry of the constellation, the distances between facial features and the assessment of each facial feature. According to the assessment of the constellations, facial feature constellations are ranked and the best constellation is chosen (Fig. 3b). Incomplete face constellations are considered as well. The contour of the connected component is considered as face contour (Fig. 3b).

2.3. Results for face detection

To assess the robustness of our approach for face detection, we applied our method to the European ACTS project M2VTS database. The database includes 37 different faces containing features like beard, glasses and different facial expressions.

The segmentation of faces fails only in one case. For facial feature extraction we obtain the results shown in Table 1. Because the faces of the database contain much more detail than the example shown in Fig. 1a, it is possible to extract also eyebrows, nostrils and chin.

features	detected [%]	correctly [%]	falsely [%]
eyebrows	81	83	17
eyes	94	74	26
nostrils	94	97	3
mouth	92	97	3
chin	86	97	3

Table 1. Detection rates for facial features

These results are rather satisfactory. Detection errors for eyebrows and eyes tend to correlate with each other. Results for face segmentation and facial feature detection are shown in Fig. 4. Eyebrows are represented by two horizontal line segments, eyes by crosses, nostrils by squares, the mouth by an horizontal line segment and the chin by an elongated rectangle.

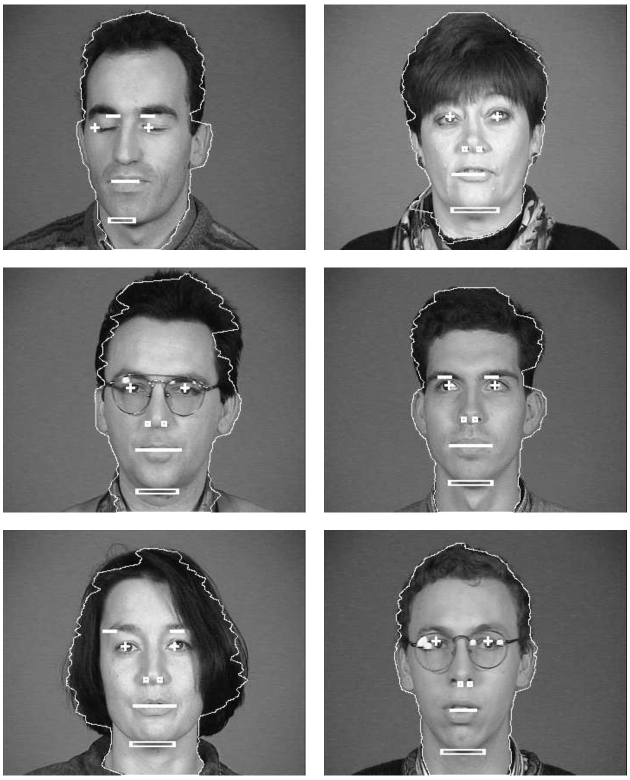


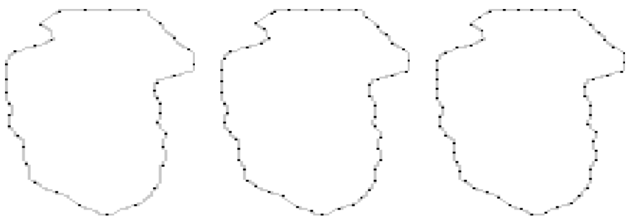
Figure 4. Results for facial feature detection

3. Tracking of the face contour

If a face is reliably detected, it can be tracked over time. A very efficient method for tracking face contours are active contours, also commonly known as snakes. An active contour is a deformable curve or contour, which is influenced by its interior and exterior forces. The interior forces impose smoothness constraints on the contour and the exterior forces attract the contour to significant image features. In a first step the snake has to be initialized and then the contour of the object is tracked by placing the snake of image f_t on image f_{t+1}. By minimizing the energy of the snake the best position of the snake in image f_{t+1} is determined.

3.1. Initilization of the snake

As described in section 2.1, we extract the face contour automatically. Thus also our snake initilization is done without interaction. Initializing the snake means to sample the face contour at time t into M nodes $v_i = (x_i, y_i)$, $i = 0, .., M-1$, also called snaxels (snake elements). There are different alternatives for the choice of the initial position of snaxels. E.g. the angle or the number of contour points or the Euclidean distance between successive snaxels can be fixed to be constant. In the case of the angle, a problem arises on contour parts with high curvature where less points are selected than in contour parts with small curvature (Figure 5a).



(a) (b) (c)

Figure 5. Snake initilization: (a) constant angle, (b) constant number of contour points, (c) constant Euclidean distance.

Furthermore, the distance between neighboured snaxels depends on the distance to the center. If the number of contour points between snaxels is fixed to be constant, these problems are solved, but when the contour runs diagonally on the image grid, the distances between snaxels increase

(Figure 5b). A proper solution for snake initialization is the use of the Euclidean distance (Fig. 5c).

3.2. Energy of snake

An active contour is an energy minimizing curve. In the discrete case the energy of a snake is defined by

$$E_{snake} = \sum_{i=0}^{M-1} E_{int}(v_i) + E_{ext}(v_i) \tag{2}$$

where E_{int} and E_{ext} denotes the interior and exterior energy terms of the snaxels v_i. The dynamic behaviour of the snake depends on the definition of these energy terms. It is necessary to choose these terms carefully to obtain a stable snake behaviour. By minimizing the energy of the snake, its optimal position is determined.

3.3. Interior energy

The interior energy makes the snake resistant to stretching and bending and ensures a smooth behaviour of the snake. In the discrete case, it is defined as follows

$$E_{int}(v_i) = w_1 \cdot \left|\frac{dv_i}{ds}\right|^2 + w_2 \cdot \left|\frac{d^2v_i}{ds^2}\right|^2 \tag{3}$$

where $v_i = (x_i, y_i)$, $i = 0, .., M-1$. The first-order term $w_1 \cdot \left|\frac{dv_i}{ds}\right|^2$ makes the snake behave like a string. A behaviour like a rod is achieved by the second-order term $w_2 \cdot \left|\frac{d^2v_i}{ds^2}\right|^2$. The first-order and second-order derivatives can be approximated by finite differences. Thus we obtain

$$\left|\frac{dv_i}{ds}\right|^2 \approx 0.5 \cdot \left(|v_i - v_{i-1}|^2 + |v_i - v_{i+1}|^2\right) \tag{4}$$

$$\left|\frac{d^2v_i}{ds^2}\right|^2 \approx |v_{i-1} - 2 \cdot v_i + v_{i+1}|^2 . \tag{5}$$

The weights w_1 and w_2 regulate the tension and rigidity of the snake. The choice of these weights is critical and it is discussed in detail in [7]. In order to mimic their physical significance, w_1 should be defined as a function of the distance between snaxels and w_2 as a function of the local curvature of a snaxel. However, it is expensive and not trivial to compute and approximate the curvature in the discrete case. Thus Leymarie and Levine recommend to fix w_2 to a small positive constant. In our approach we have decided to define these weights as functions in dependence on predefined values for a natural distance and natural curvature between snaxels. In case of w_1, we choose the initial Euclidean distance d_{init} as natural distance. The resulting function of w_1 is illustrated in Figure 6a. The weighting factor w_1 has the value 0, if the distance between snaxels is similar to the initial Euclidean distance. Thus, the first-order term of the interior energy becomes 0 and no costs for stretching arise. In case that the distance between snaxels is very large or very small, w_1 has the value 1.0 and the full costs for stretching arise.
Because the local curvature of an ellipse varies between 0 and a maximal value c_{max}, we define the natural curvature as interval $[0, c_{max}]$. In general, c_{max} depends on the ratio of minor and major axis of an ellipse, but we choose this value as a constant that is appropriate for the elliptical shape of the face. The resulting function of w_2 is shown in Figure 6b.

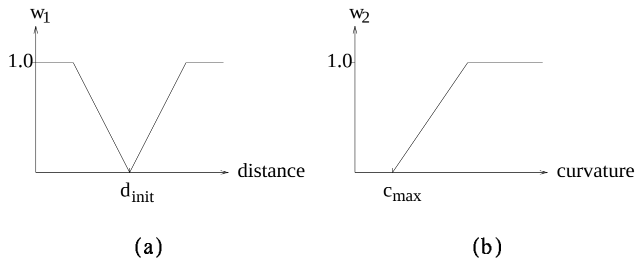


Figure 6. Weighting functions (a) w_1 and (b) w_2.

3.4. Exterior energy

The exterior energy arises from the force that pulls the snake to the significant image features. The features of interest depend on the application. Often edge information is used to adapt the snake to the contour of an object. In our case we consider color features that push or pull snaxels perpendicular to the snake.
Because the face is more or less a connected component with skin color, we check for every snaxel if the pixels in its neighbourhood have skin color or not. Pixels with skin color that are outside of the snake pull snaxels outside, pixels with no skin color that are inside of the snake push snaxels inside (Figure 7a). Thus we define the exterior energy term of a snaxel as follows:

$$E_{ext}(v_i) = -\sum_{(x,y)\in N_{int}(v_i)} 1 - s(x,y) + \sum_{(x,y)\in N_{ext}(v_i)} s(x,y) \tag{6}$$

Hereby $s(x, y)$ denotes the indicator function of skin color which is defined on the basis of the color attributes hue and saturation as described in section 2.1. N_{int} and N_{ext} are the interior and exterior neighbourhood of a snaxel (Figure 7b). The directions of these neighbourhoods are computed perpendicular to the contour direction at the snaxel position. The size of N_{int}, respectively N_{ext}, is fixed to be 5×8 pixels.

The behaviour of these exterior forces is similar to balloon forces that are decribed in [4].

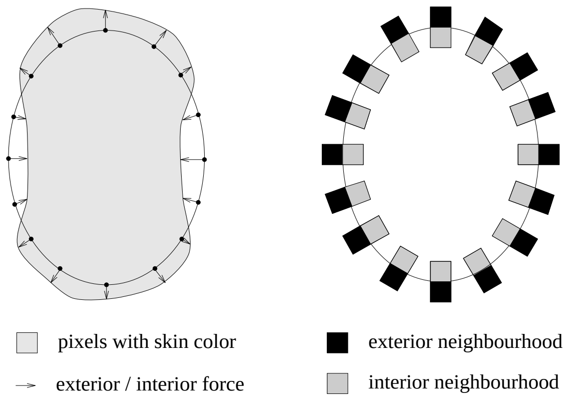


Figure 7. (a) Exterior forces on snaxels and (b) neighbourhood of snaxels.

3.5. Energy minimization

The interior and exterior forces of the snake are in balance, if the energy of the snake E_{snake} is minimal. In this case, a stable energy state is reached. The minimization of E_{snake} can be done by using the Euler-Lagrange equations of motion. A solution is obtained by using finite differences (e.g. [7]) or finite elements (e.g. [4], [9]). An alternative method for determining the energy minimum of a snake is the Greedy algorithm ([8], [14]). We use it in our approach, because of its low computational costs. It uses the fact that the energy of the snake decreases, if the energy of the snaxels decreases. Thus, the minimization problem is solved locally. Each snaxel is moved in its 8-neighbourhood and for each position the sum of interior and exterior energy is determined. The new position of the snaxel is determined based on the local energy minimum. The interior energy term of a snaxel ensures that the snake is not stretched or bended too much. This process is iterated and, after each iteration, E_{snake} is computed. The process stops, when E_{snake} converges.

The main advantage of the Greedy algorithm is its low computational costs. However by solving the minimization problem locally, the risk arises that snaxels get stuck in local minima. In our case this risk is very small, because we evaluate the color characteristics for each snaxel in a large neighbourhood and thus the exterior forces pull or push the snaxel to a significant minimum.

3.6. Addition and deletion of snaxels

In order to obtain a high variability of the snake, we allow the addition and deletion of snaxels. For example in case that a rigid object moves away from the camera, its size decreases and less snaxels are necessary for tracking. Obviously, also in the case of tracking non-rigid objects a variable number of snaxels is of advantage. The decision, whether snaxels should be added or deleted, is taken in dependence on the inter-snaxel distance. In the case that the measured Euclidean distance between two snaxels is lower a minimal threshold, one of them is deleted. If the inter-snaxel distance exceeds a maximal threshold, a new snaxel is added.

3.7. Results for contour tracking

The presented tracking algorithm was tested successfully on a number of image sequences. An example on an image sequence having 150 frames is shown in Figure 8. Though the person moves its head very much, we succeed to track it over the entire sequence. Results for every tenth frame are shown in Figure 8.

4. Conclusion

We have presented an approach for automatically segmentation and tracking of faces in color images. The segmentation of faces is done based on color and shape information. By searching for facial features, face hypotheses are verified. Afterwards tracking is performed by using an active contour model. Results for tracking were shown for an image sequence consisting of 150 frames.

References

[1] M. Acheroy, C. Beumier, J. Bigün, G. Chollet, B. Duc, S. Fischer, D. Genoud, P. Lockwood, G. Maitre, S. Pigeon, I. Pitas, K. Sobottka, and L. Vandendorpe. Multi-modal person verification tools using speech and images. In *European Conference on Multimedia Applications, Services and Techniques*, pages 747–761, Louvain-La-Neuve, Belgium, May 28-30 1996.

[2] M. Bichsel, editor. *International Workshop on Automatic Face and Gesture Recognition*, Zurich, Switzerland, June 26-28 1995. IEEE Computer Society, Swiss Informaticians Society, Swiss Computer Graphics Association et al., MultiMedia Laboratory, Department of Computer Science, University of Zurich.

[3] R. Chellappa, C. Wilson, and S. Sirohey. Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, 83(5):705–740, May 1995.

[4] L. D. Cohen and I. Cohen. Finite-element methods for active contour models and balloons for 2-D and 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1131–1147, November 1993.

[5] Y. Dai and Y. Nakano. Extraction of facial images from complex background using color information and SGLD matrices. In *IWAFGR*, pages 238–242, Zurich, Switzerland, June 26-28 1995.

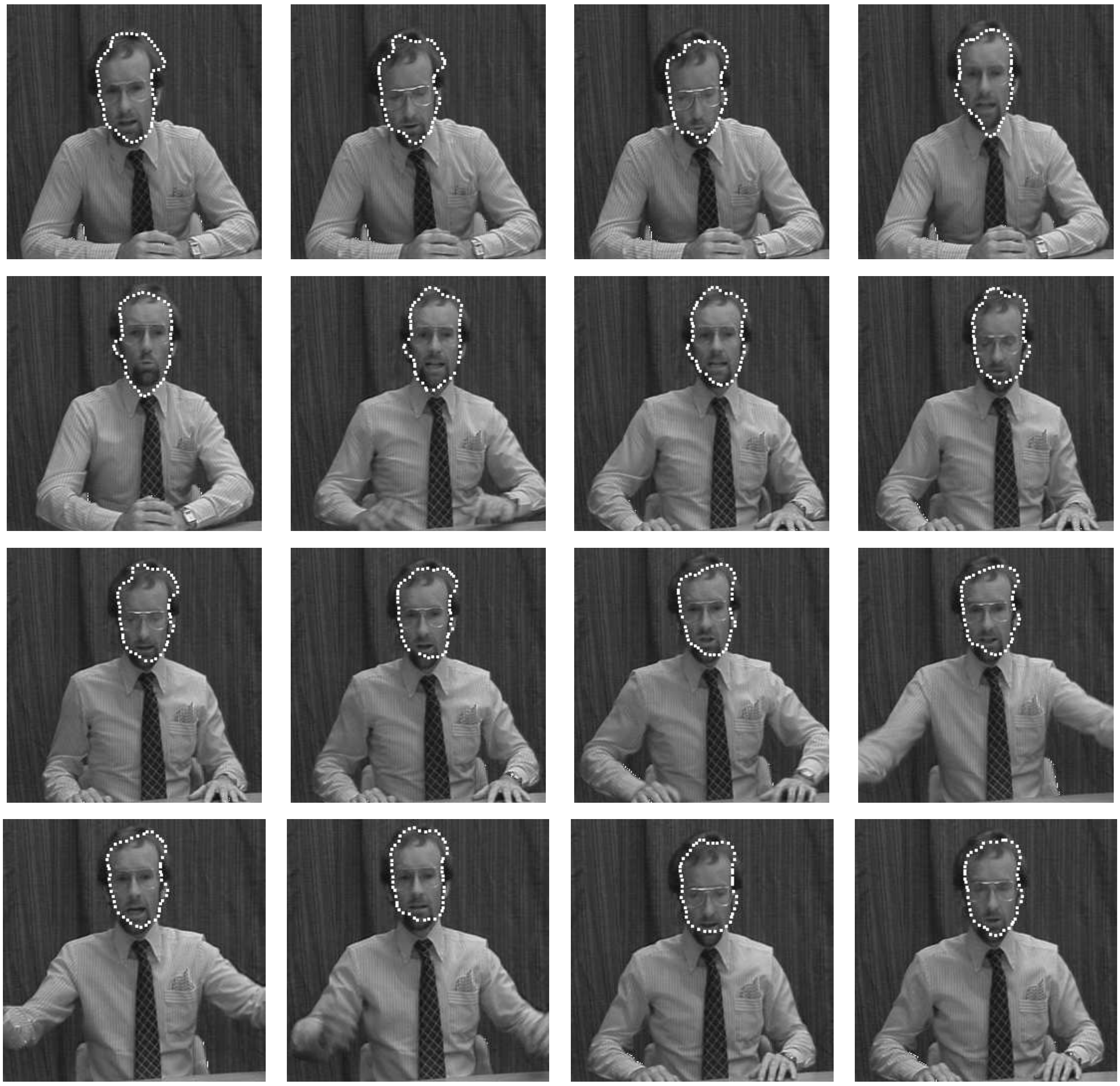


Figure 8. Face contour tracking

[6] A. Eleftheriadis and A. Jacquin. Automatic face location, detection and tracking for model-assisted coding of video teleconferencing sequences at low bit-rates. *Signal Processing: Image Communication*, 7(3):231–248, Jul 1995.

[7] F. Leymarie and M. D. Levine. Tracking deformable objects in the plane using an active contour model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):617–634, June 1993.

[8] U. Pautz. Active contours for 3D-segmentation of magnetic resonance images (in German). Diploma Thesis, FORWISS, Erlangen, Germany, July 1995.

[9] A. Pentland and S. Sclaroff. Closed-form solutions for physically based shape modeling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):715–729, July 1991.

[10] M. Reinders, P. van Beek, B. Sankur, and J. van der Lubbe. Facial feature localization and adaption of a generic face model for model-based coding. *Signal Processing: Image Communication*, 7(1):57–74, Jul 1995.

[11] K. Sobottka and I. Pitas. Extraction of facial regions and features using color and shape information. In *Int. Conf. on Pattern Recognition*, Vienna, Austria, August 1996.

[12] K. Sobottka and I. Pitas. Face localization and facial feature extraction based on shape and color information. In *Int. Conf. on Image Processing*, Lausanne, Switzerland, September 1996.

[13] S. Tominaga. Color image segmentation using three perceptual attributes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 628–630, 1986.

[14] D. Williams and M. Schah. A fast algorithm for active contours and curvature estimation. *CVGIP: Image Understanding*, Bd. 55(1):14–26, Jan 1992.