

Face Analysis using Curve Edge Maps

Francis Deboeverie¹, Peter Veelaert² and Wilfried Philips¹

¹ Ghent University - Image Processing and Interpretation/IBBT,
St-Pietersnieuwstraat 41, B9000 Ghent, Belgium
Francis.Deboeverie@telin.ugent.be,
Wilfried.Philips@telin.ugent.be,

² University College Ghent - Engineering Sciences,
Schoonmeersstraat 52, B9000 Ghent, Belgium
Peter.Veelaert@hogent.be

Abstract. This paper proposes an automatic and real-time system for face analysis, usable in visual communication applications. In this approach, faces are represented with Curve Edge Maps, which are collections of polynomial segments with a convex region. The segments are extracted from edge pixels using an adaptive incremental linear-time fitting algorithm, which is based on constructive polynomial fitting. The face analysis system considers face tracking, face recognition and facial feature detection, using Curve Edge Maps driven by histograms of intensities and histograms of relative positions. When applied to different face databases and video sequences, the average face recognition rate is 95.51%, the average facial feature detection rate is 91.92% and the accuracy in location of the facial features is 2.18% in terms of the size of the face, which is comparable with or better than the results in literature. However, our method has the advantages of simplicity, real-time performance and extensibility to the different aspects of face analysis, such as recognition of facial expressions and talking.

Keywords: geometric features, shape modeling, shape matching, face recognition, face tracking, facial feature detection.

1 Introduction

Automatic, reliable and fast face analysis is becoming an area of growing interest in computer vision research. This paper presents an automatic and real-time system for face analysis, which includes face recognition, face tracking and facial feature detection, and may be used in visual communication applications, such as video conferencing, virtual reality, human machine interaction, surveillance, etc.

The task of facial analysis has generally been addressed by algorithms that use shape and texture modelling. The Active Shape Model (ASM), proposed by Cootes et. al. [1], is one of the early approaches that attempts to fit the data with a model that can deform in ways consistent with a training set. The Active Appearance Model (AAM) [2] is a popular extension of the ASM. AAM is an integrated statistical model which combines a model of shape variation with a model of the appearance variations in a shape-normalized frame. The recently proposed Boosted Appearance Model (BAM), proposed by Liu et. al. [3,4], uses a shape representation similar to AAM, whereas

the appearance is given by a set of discriminative features, trained to form a boosted classifier, able to distinguish between correct and incorrect face alignment. Liang et. al. [5] proposed a component-based discriminative approach for face alignment without requiring initialization. A set of learned direction classifiers guide the search of the configurations of facial components among multiple detected modes of facial components. Many of the methods mentioned are complex, dependent from initialization and not always capable to handle the different aspects of face analysis.

We propose face analysis with a compact feature, the Curve Edge Map (CEM), which describes polynomial segments with a convex region in an edge map. The face CEM is a simple and natural description, which still preserves sufficient information about the facial position and expression. In fact, many curve segments correspond to physically meaningful features such as eyebrows, cheekbones or lips. The CEM approach not only has the advantages of geometric feature-based approaches, such as low memory requirements, but also has the advantage of high recognition performance of template matching.

In this paper, we employ non-local matching of curve segments in different CEMs to find corresponding curve segments. For matching, the curve segments do not have to be in each others neighbourhood. The technique differs from the matching technique in [6], where a local matching technique is described using distance and intensity characteristics. The latter technique requires that CEMs are aligned. The matching technique we propose here, is more general and does not require that the CEMs are aligned. It considers three characteristics of the curve segments.

- The first characteristic is the orientation of the main axis of the curve segment to make a first binary classification.
- The second characteristic is the intensity difference between the inner and outer side of the curve segment. Each curve segment defines a convex region, so that it makes sense to distinguish curve segments from facial features with intensity histograms between the inner and outer side.
- The third characteristic is the relative position of the curve segment in the face CEM, found by the characteristic positions of the the curve segments from the facial features in the face CEMs. This topology is modeled by a histogram of relative positions in log-polar space, which is based on the work of shape contexts from Belongie et. al. [7].

The three characteristics classify an individual curve segment in the face CEM, which is useful for facial feature detection.

We give a brief overview of three tasks handled in our face analysis system, which will be explained in the following sections:

- *Face tracking*: The face CEMs in consecutive frames are matched using the non-local matching technique as mentioned above. The motion vectors from the corresponding curve segment pairs are constructed as described in [9], in which vehicles are succesfully tracked with parabola segments. Note that faces are first detected using the cascade-based face detection algorithm of Viola and Jones [8].
- *Face recognition*: The input face CEM is matched with the face CEMs from a database using the non-local matching technique. From the matching curve segment

pairs, a global matching cost is computed. A face is recognized when the global matching cost reaches a minimum.

- *Facial feature detection:* The curve segments from facial features, such as the eyebrows, the eyes, the nose and the lips, are detected in frontal face images, by matching the face CEMs with curve segment models. These curve segment models are built for each facial feature by a training process and consist of the three characteristics as mentioned above.

Our method performs well when evaluated over different databases and video sequences, considering variations in scale, lighting, facial expressions and pose. The considered public available databases are the Georgia Tech Face Database [10], the Database of Faces [11], the Bern University Face Database [12], the AR Face Database [13], the Yale University Face Database [14] and the BioID Database [15] with ground truth marker points.

We achieve an average face recognition rate of 95.51%, which is better than the results for the technique described in [6]. Furthermore, we gain the advantage of facial feature detection, which automatically involves facial action recognition, such as recognition of facial expressions, talking and head movement. When applied to different test databases, the average facial feature detection rate is 91.92%. In this paper, we compare our results with the work of Cristinacce et. al. [16], in which they use a Constrained Local Model (CLM) to locate a set of feature points, initialized by facial feature locations found by applying the Viola and Jones face detector [8].

The accuracy in location of our facial features is computed for the BioID Database by comparing the ground truth marker points with the facial features detected. When using the mean euclidean distance as the basic error measurement, the mean location error is 6.53 pixels with standard deviation 1.15 pixels, which is 2.18% in terms of the size of the face. In this paper, we compare our results to the work of Ding et. al. [17], in which they achieve facial feature detection by learning the textural information and the context of the facial feature to be detected.

The face analysis results are comparable with or better than existing methods, when taking into account the limitations of image scale and lighting. However, the advantages of our method are simplicity, real-time properties and many face analysis tasks are handled by the same approach.

This paper is organized as follows. First, we briefly indicate how to compute a CEM for a gray level image in Section 2. In the following Section 3, we present the basic parts for matching the curve segments. Section 4 describes in detail the different applications of the face CEM. In Section 5 we show the results when testing our face analysis system on different databases and video sequences. Finally, we conclude our paper in Section 6.

2 Curve segmentation

The basic step of our method is the interpretation and description of an image edge map by geometric primitives. Our feature representation, the Curve Edge Map (CEM), integrates structural information and spatial information by grouping pixels of an edge map into curve segments. In [6,18], it is shown that the approximation of edge maps by

second order polynomials or parabola segments is a useful and elegant representation for both non-rigid and rigid structures.



Fig. 1. Curve segmentation: Images (a), (b) and (c) show the input image, the edge map and the face CEM, respectively.

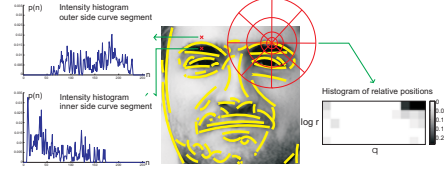


Fig. 2. Histograms of intensities and relative positions: On the left are shown the intensity histograms from the inner and outer side of the curve segment, which describes the left eyebrow. On the right is plotted the log-polar histogram of relative positions from the curve segment, which describes the right eyebrow.

The edge map is computed with the Canny edge detector [19], a frequently used edge detector when considering edge information in faces [20], resulting in thin edges of one pixel thickness. Digitized curves are obtained from the edge map by a simple boundary scan algorithm. To fit curve segments, we use an adaptive incremental linear-time fitting algorithm for curve segmentation which is based on constructive polynomial fitting [21,18]. The output of the fitting algorithm is a list of curve segments that approximates the edge map with an L_∞ user-specified threshold. In practice, we represent the curve segments as parabola segments, represented by three parameters, which have their main axes in x or y direction, depending on which direction yields the smallest fitting cost. The restriction of allowing only two orientations is based on the many vertical and horizontal orientations of the facial features in a face. Figures 1 (a), (b) and (c) show the input image, the edge map and the CEM for a face from the Bern University Face Database [12], respectively.

3 Curve Edge Map matching

3.1 Curve coefficients classification

The orientations of the curve axes are used to make a first binary classification, by which we mean that the curve segments will only match when their main axes have the same orientation. For example for the lips and the eyebrows the alignment of the curve axes are vertical, while for the nose the alignment of the curve axis is horizontal. In practice, we consider the axes of symmetry of the parabola segments, which are in the x or the y direction.

3.2 Curve intensity histograms

One of the discriminative properties in our system is the difference in intensity between the inner and outer side of the curve segment. For each side we construct a normalized intensity histogram.

When estimating a histogram from one side of the curve segment, the region of interest for intensities is in the area between the original curve segment and a duplicate which is translated parallel to the main axis of the curve segment. The distance d over which the duplicate is translated results in a robust face recognition rate R , when d is between 0.02 and 0.09 of the square root of the face surface A , where A is the rectangle produced by face detector. This optimization is done for the AR Face Database [13].

Figure 2 shows on the left two intensity histograms from the inner and outer side from the curve segment of the left eyebrow for a face from the Bern University Face Database [12]. The histogram representing the upper side has its intensities on the bright side, while the histogram representing the lower side has its intensities on the dark side.

To match the intensity histograms from one side of two different curve segments, we use the Bhattacharyya distance metric or B-distance measure, which measures the similarity of two probability distributions and is a value between 0 and 1. This measure is chosen for its good classification properties [22]. The B-distance between two intensity histograms f and g is

$$B(f, g) = 1 - \sum_{l=0}^L \sqrt{f(l)g(l)}, \quad (1)$$

where L is the number of bins in the histograms, $L = 255$ for gray images. The matching of intensity histograms is done for the inner and outer side of the curve segments, resulting in B_{in} and B_{out} .

3.3 Histograms of relative positions in CEM

A curve segment in the CEM is characterized by its relative position and becomes important when classifying individual facial features, for example to distinguish between curve segments from the eyebrow and the upper lip, which have similar transitions in intensities between inner and outer side. In our work, the topology of the face CEM is modeled using shape contexts. In the original shape context approach [7], a shape is represented by a discrete set of sampled points $P = p_1, p_2, \dots, p_n$. For each point $p_i \in P$, a coarse histogram h_i is computed to define the local shape context of p_i . To ensure that the local descriptor is sensitive to nearby points, the local histogram is computed in a log-polar space. In our case, a histogram is computed for the center point of the curve segment in the face CEM. When considering such a center point, the sampled points P are the discretized points on the other curve segments in the face CEM. An example of a histogram of relative positions is shown in Figure 2, on the right is plotted the log-polar histogram from the curve segment of the right eyebrow. In practice the circle template covers the entire face.

Assume that p_i and q_j are the center points of the curve segments of two different faces. The shape context approach defines the cost of matching the two curve segments by the following χ^2 test statistic:

$$C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^K \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)} \quad (2)$$

where $h_i(k)$ and $h_j(k)$ denote the K-bin normalized histograms of relative positions of p_i and q_j , respectively.

4 Applications with face CEM

4.1 Face tracking

The face CEMs in consecutive frames are matched using the technique as described in Section 3. While the orientations of the curve axes are used to make a first binary classification, the costs for matching intensity histograms and matching histograms of relative positions are linearly combined for reasons of simplicity. Experiments show that for a pair m of matching curve segments, the linearly combined cost $D_m = a(B_{in} + B_{out})/2 + bC$, where B_{in} and B_{out} are costs for matching intensity histograms and C is the cost for matching histograms of relative positions, has a maximized recognition rate when $a = 0.83$ and $b = 0.17$.

We look for a one-to-one match for each of the curve segments in the current frame by a minimization of the linearly combined cost D_m . For the construction of motion vectors we use similar techniques as proposed in [9]. A motion vector of a unique curve correspondence pair is defined by the center points of two curve segments.

4.2 Face recognition

The input face CEM is matched with the face CEMs in a database using the technique described in Section 3. From the pairs of matching curve segments, a global length weighted matching cost E is computed, with $E = \sum_{m=1}^M D_m l_m / \sum_{m=1}^M l_m$, where M is the number of unique pairs of curve segments and l_m is the average length of the matching curve segment pair m . Then, a face is recognized when the global matching cost reaches a minimum.

4.3 Facial feature detection

Facial features such as the eyebrows, the eyes, the nose and the lips, are detected using the face CEMs.

Firstly, curve segments from the facial features of interest have been modeled by a training process on a database, consisting of 100 faces, considering small variations in pose, variations in lighting conditions and variations in facial expressions. The curve segment model of a facial feature consists of an orientation of the axis of the curve segment, two intensity histograms from the inner and outer side of the curve segment and a log-polar histogram describing the relative positions in the face CEM.

Secondly, the curve segment models are matched with the input face CEM. The coefficients of the curve segments make a first binary classification. For the lips and the eyebrows the alignment of the curve segment axes are vertical, while for the nose the alignment of the curve segment axis is horizontal. Next, the histograms of intensities and the histograms of relative positions are compared. E.g. for the upper curve segment of a left eyebrow we expect a transition in intensity from bright to dark and with a relative position in the left upper corner in the face CEM. In this manner, we classify the curve segments from the left and right eyebrow, the left and right eye, the left and right side of the nasal bone and the upper and the lower lip.

5 Results

Our system applications as described in Section 4 are evaluated on different face databases, namely the Georgia Tech Face Database [10], the Database of Faces [11], the Bern University Face Database [12], the AR Face Database [13], the Yale University Face Database [14] and the BioID Database [15] with ground truth marker points.

To show the real-time properties of our system, we give in Table 1 an overview of the computational performance of our face analysis system. The system is implemented in C++ as a webcam application, running on a 2.80 GHz processor, 4.00 GB RAM and 64-bit operating system. We average computational time over 1000 frames for the face detection, the CEM construction and the CEM matching, considering different frame and face sizes.

The CEM matching is evaluated by the application of face recognition as described in section 4.2. The face recognition rates are given in Table 3. A distinction is made between the matching technique described in [6], matching using histograms of relative positions, matching using intensity histograms and matching using an optimal linear combination of the latter two. In the results, the correct match is only counted when the best matched face from a model is the correct person. We test face recognition under controlled condition and size variation, under varying lighting condition, under varying facial expression and under varying pose. As we expect, the matching of histograms with relative positions is more sensitive to varying pose and varying facial expressions, while the matching of histograms is more sensitive to varying lighting conditions. The average face recognition for the linearly combined cost is 95.51%. When compared to the results described for the matching technique in [6], the average face recognition rate increases with 2.21%. Furthermore, we gain the advantage of facial feature detection, which involves facial action recognition, such as recognition of facial expressions, talking and head movement.

The facial feature detection as described in Section 4.3 is evaluated on the test databases by verifying whether or not the curve segment models classify the correct curve segments in the faces. We classify the curve segments from the left and the right eyebrow, the left and the right eye, the left and the right side of the nasal bone and the upper and the lower lip, as shown in Figures 3 (a) and (b). The results for facial feature detection, as presented in Table 2, show that the developed system can detect on average the individual facial features successfully in 91.92% cases, when applied to the test databases. As a comparison, our average facial feature detection rate for the BioID

Frame size	320x240	640x480	1280x960
Face detection (ms)	48.85	52.27	102.93
Face size	100x100	200x200	400x400
CEM construction (ms)	16.35	37.88	61.38
CEM matching (ms)	9.16	26.36	43.38
Total (ms)	74.36	116.51	207.69
fps	13.45	8.58	4.81

Table 1. Computational performance: Average computational time for face detection, CEM construction and CEM matching, considering different frame and face sizes.

(%)	YFD	BERN	AR	GTFD	ATT	BioID
Left eyebrow	100.00	90.00	87.91	96.00	92.50	93.12
Right eyebrow	100.00	80.00	90.47	94.00	90.00	94.65
Left eye	93.33	90.00	86.20	86.00	85.00	91.39
Right eye	93.33	93.33	87.91	88.00	87.50	90.94
Left nose	100.00	90.00	88.03	96.00	92.50	94.10
Right nose	100.00	96.67	87.18	90.00	95.00	92.94
Upper lip	93.33	90.00	89.20	86.00	97.50	95.51
Lower lip	100.00	96.67	90.60	86.00	92.50	93.77
Average	97.50	90.46	88.44	90.25	91.56	93.30

Table 2. Results for facial feature detection: The detection rates for the left and the right eyebrow, the left and the right eye, the left and the right side of the nasal bone and the upper and lower lip.

(%)	Old [6]	Hist. of rel. pos.	Intensity hist.	Lin. comb.
<i>Controlled condition</i>				
GTFD [10]	98.00	82.00	98.00	100.00
ATT [11]	100.00	77.50	95.00	98.50
YFD [14]	NA	86.66	100.00	100.00
BERN [12]	100.00	93.33	96.67	100.00
AR [13]	98.00	78.65	96.02	98.44
<i>Varying pose</i>				
BERN Right	93.34	70.33	87.00	91.33
BERN Left	86.67	70.67	88.33	91.67
BERN Up	86.67	72.00	86.67	90.33
BERN Down	73.34	69.00	80.33	83.67
<i>Size variation</i>				
AR with size variation	90.21	77.83	89.20	94.35
<i>Varying lighting condition</i>				
AR with left light on	96.34	73.53	91.89	96.60
AR with right light on	94.84	73.43	89.16	95.63
AR with both lights on	92.10	75.15	90.63	96.38
<i>Varying facial expression</i>				
AR with smiling expr.	96.53	71.13	93.70	97.13
AR with angry expr.	97.30	71.59	94.14	97.46
AR with screaming expr.	96.21	72.08	91.40	96.65
Average	93.30	76.18	91.76	95.51

Table 3. Results for face recognition: The second, third, fourth and fifth columns show the recognition rates for the matching technique described in [6], matching using histograms of relative positions, matching using intensity histograms and matching using an optimal linear combination of both, respectively.

Database is 93.30%, which is comparable with the facial feature detection rate by applying the Viola and Jones face detector [8]. The Viola and Jones face detector finds 95% of facial feature points within 20% of the inter-ocular separation on the BioID Database [16]. However, experiments show that our approach has the advantages of low computational cost and large invariance for variations in facial expressions.

The accuracy in position of the facial features with the curve segments is determined by the accuracy in position of the edges delivered by the canny edge detector and the fitting cost allowed during segmentation. We compute this accuracy on the BioID Database, by comparing the available ground truth marker points with the locations of the facial features detected. In Figure 3 (a), the ground truth marker points are indicated with red crosses. The criteria for the accuracy is the distance of the points on the curved segments closest to the ground truth markers. When using the mean euclidean distance as the basic error measurement, the mean detection error is 6.53 pixels with a standard deviation of 1.15 pixels, which is 2.18% in terms of the size of the face. We compare to the work of Ding et. al. [17], in which they compute the average accuracy on the AR Face Database and the XM2VT Face Database. The mean detection error for SubAdaBoost is 9.0 pixels with a standard deviation of 1.3 pixels, which is 2.8% in terms of the size of the face.

As a result for face tracking, Figures 4 (a), (b), (c) and (d) show the tracking of a face, which is looking left, right, up and down, respectively. From the motion vectors

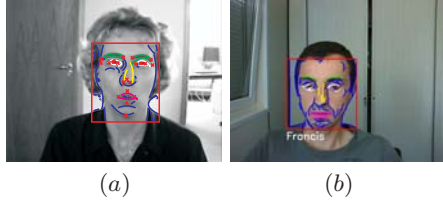


Fig. 3. Detection of facial features: Image (a) shows the curve segments of the eye-brows, the eyes, the nose and the lips detected in a face from the BioID Database [15]. The ground truth marker points are indicated with red crosses. Image (b) shows the facial feature detection in a face from a webcam video sequence.

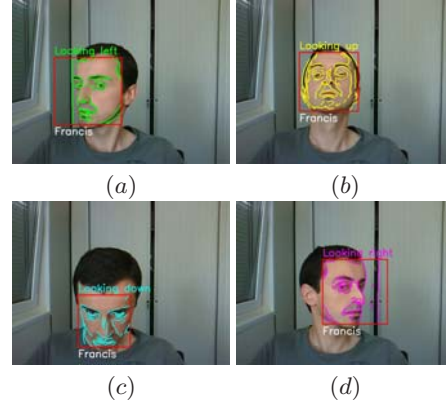


Fig. 4. Results for face tracking: Image (a), (b), (c) and (d) show a face from a webcam video sequence, which is looking left, right, up and down, respectively.

of the matching parabola segment pairs, an average motion vector makes an estimation about the head movement. In future work, facial action recognition, such as facial expression recognition, head movement recognition and talking recognition will be further explored.

6 Conclusion

This paper proposes an alternative way to the analysis of faces. Our method models faces with Curve Edge Maps, which is a natural description for the many edges in a face and correspond to physically meaningful features, such as the eyebrows, the eyes, the nose and the lips. We presented a novel matching technique, which uses the orientation of the axes of the curve segments, intensity histograms and histograms of relative positions. We demonstrated that the CEM approach is useful for many face analysis tasks, such as face tracking, face recognition and facial feature detection. Applied to different databases, good performance for these applications is achieved, which is comparable with or better than the results in literature. However, our method provides simplicity, real-time performance and extensibility to the different aspects of face analysis.

References

1. T. F. Cootes and C. J. Taylor and D. H. Cooper and J. Graham, Training Models of Shape from Sets of Examples, Proc. British Machine Vision Conference, pp. 9-18, 1992.
2. T. F. Cootes and G. J. Edwards and C. J. Taylor, Active Appearance Models, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, pp. 681-685, 2001.
3. Xiaoming Liu, Generic face alignment using boosted appearance model, Proc. IEEE Computer Vision and Pattern Recognition, pp. 1079-1088, 2007.

4. Xiaoming Liu, Discriminative Face Alignment, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 1941-1954, 2009.
5. Lin Liang and Rong Xiao and Fang Wen and Jian Sun, Face Alignment via Component-based Discriminative Search, *Proc. of European Conference on Computer Vision*, pp. 72-85, 2008.
6. Francis Deboeverie and Peter Veelaert and Kristof Teelen and Wilfried Philips, Face Recognition using Parabola Edge Map, *Proc. of Advanced Concepts for Intelligent Vision Systems*, vol. 5259, pp. 1079-1088, 2008.
7. Serge Belongie and Jitendra Malik and Jan Puzicha, Shape Matching and Object Recognition Using Shape Contexts, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 509-522, 2001.
8. Paul Viola and Michael Jones, Rapid object detection using a boosted cascade of simple features, *Proc. of Computer Vision and Pattern Recognition*, pp. 511-518, 2001.
9. Francis Deboeverie and Kristof Teelen and Peter Veelaert and Wilfried Philips, Vehicle Tracking using Geometric Features, *Proc. of Advanced Concepts for Intelligent Vision Systems*, vol. 5807, pp. 506-515, 2009.
10. Georgia Institute of Technology, Georgia Tech Face Database, http://www.anefian.com/face_reco.htm.
11. AT&T Laboratories, Cambridge, The Database of Faces, <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>.
12. University of Bern, Bern, Switzerland, Bern University Face Database, <ftp://iamftp.unibe.ch/pub/ImagesFaceImages/>.
13. A.M. Martinez and R. Benavente, The AR Face Database, *CVC Technical Report #24*, 1998.
14. University of Yale, Bern, Switzerland, Yale University Face Database, <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>.
15. Oliver Jesorsky and Klaus J. Kirchberg and Robert W. Frischholz, Robust Face Detection Using the Hausdorff Distance, *Proc. Third International Conference on Audio- and Video-based Biometric Person Authentication*, pp. 90-95, 2001.
16. David Cristinacce, T. F. Cootes, Automatic Feature Localisation with Constrained Local Models, *Pattern Recognition*, vol. 41, no. 10, pp. 3054-3067, 2008.
17. Liya Ding and Aleix M. Martinez, Features versus Context: An Approach for Precise and Detailed Detection and Delineation of Faces and Facial Features, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 2022-2038, 2010.
18. Francis Deboeverie and Kristof Teelen and Peter Veelaert and Wilfried Philips, Adaptive Constructive Polynomial Fitting, *Proc. of Advanced Concepts for Intelligent Vision Systems*, vol. 6474, pp. 173-184, 2010.
19. J. Canny, A Computational Approach to Edge Detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679-698, 1986.
20. Kailash J. Karande and Sanjay N. Talbar, Independent Component Analysis of Edge Information for Face Recognition, *International Journal of Image Processing*, vol. 3, no. 3, pp. 120-130, 2009.
21. Peter Veelaert and Kristof Teelen, Fast polynomial segmentation of digitized curves, *Proc. of Discrete Geometry for Computer Imagery*, vol. 4245, pp. 482-493, 2006.
22. Sung-Hyuk Cha and Sargur N. Srihari, On measuring the distance between histograms, *Pattern Recognition*, vol. 35, no. 6, pp. 1355-1370, 2002.