Real-Time Facial Feature Detection By Statistical Template Matching

ALEXANDER SIBIRYAKOV, MIROSLAW BOBER
Mitsubishi Electric ITE-VIL,
20 Frederick Sanger Road, The Surrey Research Park, Giuldford, GU2 7YD
UNITED KINGDOM
http://www.vil.ite.mee.com

Abstract: This paper addresses a problem of robust, accurate and fast object detection in complex environments, such as cluttered backgrounds and low-quality images. A new method called Statistical Template Matching is proposed to detect objects, represented by a set of template regions. A similarity measure between the image and a template is derived from the Fisher criterion. We show how to apply our method to face and facial feature detection tasks, and demonstrate its performance in some difficult cases, such as moderate variation of scale factor of the object, local image warping and distortions caused by image compression.

Key-Words: Statistical template matching, Topological template, Object detection, Facial feature detection.

1 Introduction

Performance of object detection methods highly depends on how the object of interest is defined. If a template describing a specific object is available, object detection becomes a process of matching features between the template and the image under analysis. A few common techniques exist for template matching. In the image subtraction technique [1], the template position is determined by minimizing the dissimilarity function between the template and image regions located at various positions. Matching by correlation [2] utilizes the position of the normalized cross-correlation peak to locate the best match. The deformable template matching approach [3] is more suitable for cases where objects vary due to rigid and non-rigid deformations. In this approach, a template describes the characteristic features of an object shape. The phase correlation method [4] is based on the Fourier Shift Theorem; it computes the cross-power spectrum of the template and the image and searches for the location of the peak in its inverse.

While these methods are widely used in vision systems, they have a number of disadvantages. Object detection by template matching is generally computationally expensive and its quality and speed depends on the details of object template. For a template of size *M* x *N,* the correlation method requires O(*MN*) operations per image pixel, and therefore it may not be suitable for real-time applications. Phase correlation is fast but it requires templates of larger sizes. If the object of interest is small, the output of the phase correlation can be poorly defined. If the rough position of objects is known a priori, e.g. from tracking, the size of the search area is reduced and phase correlation performs well, but another detection method is required to initialize region tracking.

To overcome the problems with existing methods, we propose a new object detection approach based on topological templates and statistical hypotheses testing. The method is very fast; its speed is independent of the template size and depends only on the template complexity.

2 Statistical Template Matching

2.1 Definitions of image and template

We assume an image *I* to be a function of *M* coordinate variables $I(x_1,x_2,\ldots, x_M)$. The case of $M=1$ corresponds to one-dimensional function representing any signal or function, for example a pixel profile extracted from a 2D-image. $M=2$ represents the usual case of a 2D-image $I(x,y)$, while $M=3$ represents a volumetric image: voxel image, image sequence or video organised as an image stack.

The object of interest is described by a set of regions $T_0 = T_1\cup\ldots\cup T_N$, representing only the topology of the object (i.e. spatial relation of its parts) and not its radiometric properties associated with radiation, such as colour or intensity. A region T_i does not have to be contiguous. This description is called *Topological Template* or simply *Template*

2.2 Statistical hypotheses testing

We call our method *Statistical Template Matching* (STM)*,* because only statistical characteristics of the pixel groups, mean and dispersion, are used in the analysis. The similarity measure between a template and image regions is based on statistical hypothesis testing. For each pixel **x** and its neighbourhood R(**x**) two hypotheses H_0 and H_1 are considered:

H_0: R(**x**) is random (not similar to the template);
H_1: R(**x**) is similar to the template.

The decision rule for accepting one of the hypotheses H_0 or H_1 is based on testing whether the characteristics of pixel groups, defined by template regions, when template is centred at pixel **x** (T_0=R(**x**)) are statistically different from each other. Let us consider first the case of two regions: $T_0=T_1 \cup T_2$. Application of the well-known statistical *t-test* to two pixel groups leads to the following similarity measure (some equivalent transformations are skipped):

$$(t)^2 = \left(\frac{Signal}{Noise}\right)^2 = \left(\frac{Difference\ between\ group\ means}{Variability\ of\ groups}\right)^2 = \tag{1}$$

$$\frac{(m(T_1)-m(T_2))^2}{\dfrac{\sigma^2(T_1)}{|T_1|}+\dfrac{\sigma^2(T_2)}{|T_2|}} = \ldots = \frac{|T_0|\sigma^2(T_0)}{|T_1|\sigma^2(T_1)+|T_2|\sigma^2(T_2)} - 1$$

where $\sigma^2(T_i)$ is the variance of the image values in a region T_i, and $|T_i|$ designates the number of pixels inside the region T_i.

When the template is composed of three or more regions another statistical technique is used to obtain the similarity measure. This technique is called Analysis Of Variances (ANOVA), which is mathematically equivalent to the t-test, and it is used if the number of groups is more than two. Denote *Between-group variation* and *Within-group variation* as $Q_1(T_1,\ldots,T_M)$ and $Q_2(T_1,\ldots,T_M)$ respectively. They are computed using equations (2) and (3), and equation (4) defines the relation between them:

$$Q_1(T_1,\ldots,T_M) = \sum_{i=1}^{M} |T_i| m^2(T_i) - |T_0| m^2(T_0) \tag{2}$$

$$Q_2(T_1,\ldots,T_M) = \sum_{i=1}^{M} |T_i| \sigma^2(T_i) \tag{3}$$

$$|T_0|\sigma^2(T_0) = Q_1(T_1,\ldots,T_M) + Q_2(T_1,\ldots,T_M) \tag{4}$$

We use the Fisher criterion as a similarity measure (equivalent transformations, following from (2)-(4), are skipped):

$$F = \frac{Q_1/(M-1)}{Q_2/(|T_0|-M)} = \ldots = \frac{|T_0|-M}{M-1}\left(\frac{|T_0|\sigma^2(T_0)}{\sum_{i=1}^{M}|T_i|\sigma^2(T_i)} - 1\right) \tag{5}$$

After removing the constants from the expressions (1) and (5), we obtain a similarity measure of the form:

$$S(\mathbf{x}) = \frac{|T_0|\sigma^2(T_0)}{|T_1|\sigma^2(T_1) + \ldots + |T_N|\sigma^2(T_N)}, \tag{6}$$

The similarity measure (6) can be also interpreted as squared signal-to-noise ratio (SNR) with values ranging from 1 (corresponding to noise) to infinity (perfect signal).

2.3 Real-time object detection

An object detection method can be implemented based on the properties of the statistical template matching. STM is applied to each pixel *x* in the image to obtain *S(x)* and a set of statistical characteristics of the regions $\sigma^2(T_0),\ldots, \sigma^2(T_N), m(T_0),\ldots, m(T_N)$, where $m(T_i)$ is a region mean and $\sigma^2(T_i)$ is its variation. Similarity values (6) form a similarity map with values corresponds to likely object locations. The SNR-based interpretation of the similarity measure opens the possibility of thresholding the confidence map for object/non-object location classification. After thresholding, non-maxima suppression is applied to detect local maxima of the similarity map and integer coordinates of detected object centres. Fitting a polynomial surface to the similarity map in the vicinity of a local maximum gives subpixel location of the object.

Application-dependent analysis of statistics $\sigma^2(T_0),\ldots, \sigma^2(T_N), m(T_0),\ldots, m(T_N)$ helps to reduce the number of false alarms. When radiometric properties of the object regions are known in advance (for example, it is known that some of the regions are darker then the others), additional conditions, such as $m(T_i) < m(T_j)$ reject unwanted configurations.

The STM can be easily implemented to achieve real-time performance by using the well-known technique called integral images. In this implementation each template region T_i consists of union of rectangles. For such regions in 2D-images each variance value in (6) can be computed by 8*k* memory references, where *k* is a number of rectangles. The conventional way of computing $\sigma^2(T_i)$ requires $|T_i|$ memory references.

From computing region mean and variance it follows that statistics for one region, say T_N, can be derived from statistics of other regions using the fact that $T_N=T_0 \cap (T_1 \cup \ldots \cup T_{N-1})$. This optimisation can give a significant increase in speed if only a small number of regions is used (*N*=2,3) or the region T_N is complex, e.g. consists of a very large number of rectangles.

The method can be applied in a coarse-to-fine framework using a few resolutions of the image and the template. The process starts from the matching of the coarsest template in the coarsest image resolution. After extracting all possible object locations from the coarse similarity map, the process is performed only inside the region-of-interest (ROI) at the finer resolutions. In object tracking applications the method initialises ROIs in the first images of a sequence and predicts their location in the next images, thus reducing the search area for the STM.

3 Facial Feature Detection

3.1 Overview of the method

One problem with many existing eye detection methods is their high computational complexity and low detection reliability. This is usually due to algorithms relying on the search of consistent region pairs (left eye, right eye) in the image. Even in a simple image many candidate region pairs can be found and this ambiguity should be resolved by some higher-level decision rules. In our method the detection is based on a region triplet containing the left eye, between-eyes and right eye regions, which are less frequent in an image. The initial hypothesis, based on such region triplet, is further validated by the presence of other facial feature regions such as the mouth, so that eye detection becomes much less ambiguous and less time-consuming.

3.2 Design of facial feature templates

To detect facial features the STM method described in Section 2 is used. STM requires topological templates consisting of union of regions. Each template represents only the general appearance of a facial feature (see Fig.1,3). The template consists of regions, showing where the distribution of pixel values is darker or lighter. Black and white areas indicate such parts of the template. Binarization method can be used to design the templates. Fig.1a,b shows an image with a facial feature of interest (between-eyes region) and its binarization. In this example the feature of interest looks like two dark elliptical regions (Fig.1c). Due to real-time processing requirements all the regions should preferably be rectangular, which leads to further simplification of the template, as it shown in Fig.1d. Fig.2 shows examples of face detection by the STM method using the template from Fig.1d. The top row of Fig.2 shows face images together with the position of the maximum of the similarity measure. The bottom row shows the corresponding fragments of the similarity maps, computed based on the similarity measure (6). Fig.3 shows the full set of the templates used in our method. Two different templates for the between-eyes region are shown in Fig.3a,b. The horizontal facial features template, shown in Fig.3c, serves to detect closed eyes, mouth, nostrils, and eyebrows. The template in Fig.3d is specially designed for open eyes (dark pupil surrounded by a lighter neighborhood).

3.3 Facial feature detection

Facial feature detection consists of two stages: 1) low-level image processing and 2) facial feature analysis and selection.

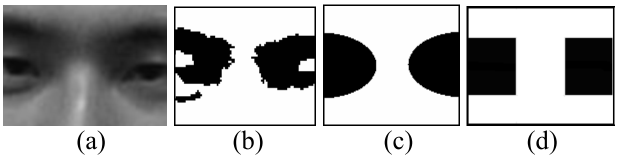
(a) (b) (c) (d)

Fig.1 Facial feature template design and simplification. (a) Original image; (b) Binarization of the Fig.1a for qualitative estimation of the template shape; (c) One possible template for detecting the face feature from Fig.1a; (d) Simplified template for real-time detection

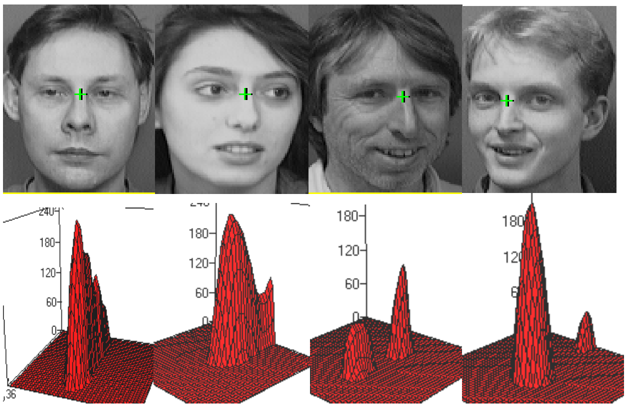

Fig.2 Application of STM to the face detection problem. Top row: face examples from the AT&T Face Database [5]. Bottom row: 3D-plot of the similarity measure (6).

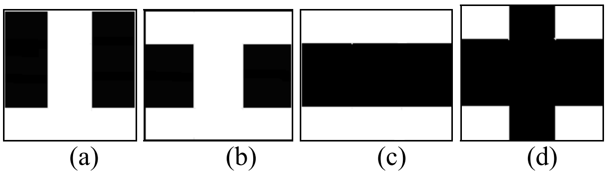
(a) (b) (c) (d)

Fig.3 Facial feature templates used in our current implementation; (a),(b) Templates for the between-eyes region; (c) Template for horizontal facial features; (d) Template for an open eye.

3.3.1 Low-level image processing stage

The low-level image processing stage starts from image transformation to integral representation so that the time required by the STM is independent of template size. Then STM is performed on a pixel-by-pixel basis, resulting in multiple confidence maps, one for each facial feature. Two maps indicate a likelihood of presence of the between-eyes region; another two maps indicate possible eye and other horizontal feature regions, such as nostrils and mouth.

These confidence maps are then combined into a final confidence map using pixel-by-pixel multiplication. Fig.4b,c show examples of combined confidence maps.

Facial feature regions with high confidence level are extracted using segmentation of the combined confidence map. Each confidence map is segmented in order to separate regions with high confidence from the background of low confidence. The threshold value of S_T=1.1, indicating that the signal level is just above the noise level, is used in the current implementation.

In order to further improve robustness we use also heuristic gradient features as weighting factors for the confidence level. Eyes and mouth usually contain sharp changes of intensity that can be used as additional evidences regarding types of the corresponding regions (Fig.5). Theses features are computed as follows:

$$G_E = \frac{1}{|R_E|}\sum_{(x,y)\in R_E}\left|f(x+1,y)-f(x,y)\right| \qquad (7)$$

$$G_M = \frac{1}{|R_M|}\sum_{(x,y)\in R_M}\left|f(x,y+1)-f(x,y)\right| \qquad (8)$$

For an eye candidate region R_E we compute the feature G_E, which is the average absolute value of horizontal derivative of the image intensity function. The horizontal component of the image gradient was chosen because neighbour face features (eyebrows) usually contain no significant edges in X-direction, but may contain edges in Y-direction (compare Fig.5a and Fig.5b). Similarly, mouth regions R_M usually contain sharp vertical changes of the intensity and the corresponding gradient measure G_M is computed using (8).

3.3.2 Facial feature analysis and selection stage

In this stage all regions with a high confidence are extracted by a connected component labelling algorithm applied to the thresholded confidence maps. Then all possible region triplets (left eye, between-eyes, right eye) are iterated and roughly checked for symmetry (Fig.4d). A region in the mouth search area, containing the highest confidence map value, is selected as a candidate for the mouth region (Fig.4f). The mouth search area is selected based on the distance between eye candidates.

In the final step of the algorithm, the triplets with high total confidence level are validated by gradient features (7),(8) and the presence of other facial feature regions such as mouth and nostrils. Positions of the global maxima of confidence level in the eye regions are considered as the exact eye positions (Fig.4f).

We denote $R_B^{\,i}$ a maximal value of the confidence map in i-th between-eyes region candidate. Similarly, we define maximal values $R_{LE}^{\,j}$, $R_{RE}^{\,k}$, $R_M^{\,l}$ for the left eye, the right eye and the mouth candidates. The corresponding gradient measures computed by (7),(8) are $G_{LE}^{\,j}$, $G_{RE}^{\,k}$, $G_M^{\,l}$. The total score for accepting the set of between- eyes, eyes and mouth region candidates as a valid facial feature set is computed as a sum of confidence values weighted by sum of the gradient measures:

$$(i^*, j^*, k^*, l^*) = \arg\max_{(i,j,k,l)}\left\{\left(R_B^i + R_{LE}^j + R_{RE}^k + R_M^l\right)\left(G_{LE}^j + G_{RE}^k + G_M^l\right)\right\} \qquad (9)$$

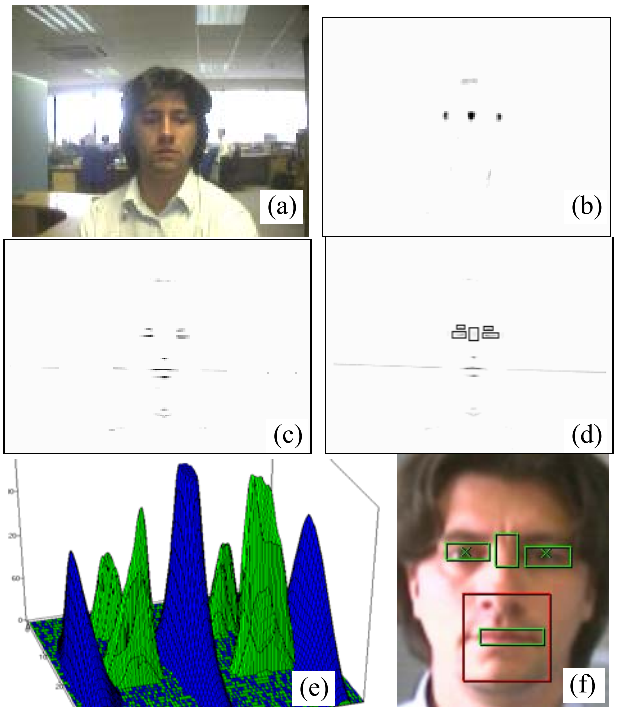

Fig.4 STM-based facial feature detection. (a) Image example; (b) Result of STM for the between-eyes region (combined confidence map for the templates from Fig.3a,b); (c) Result of STM for the eye region (combined confidence map for the templates from Fig.3c,d); (d) Fig.4b,c are combined and valid (left eye, between-eyes, right eye) triplets are outlined; (e) 3D-plot of the combined confidence map from Fig.4d; it shows that correct triplet has the largest sum of similarity values; (f) Final steps of the detection algorithm. The region set having highest score (9) is shown. The mouth region is selected from the outlined search area. The eye regions are shown together with the eye positions.

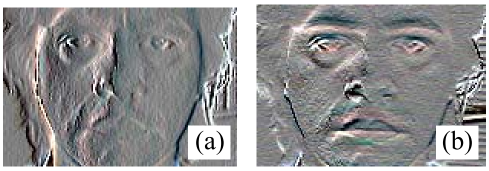

Fig.5 Computation of the gradient features. (a) A face image convolved with the [-1 1] mask; (b) The image convolved with the [-1 1]^T mask.

3.4 Multi-resolution approach

In our implementation, the detection algorithm is applied first to the downsampled versions of the image and templates. This significantly reduces the computational time, but also may reduce the effectives and accuracy of facial feature detection. Often eyes are more easily detected in the downsampled images, because some confusing details, such as glasses, disappear at the reduced resolution. However, the opposite situation is also possible: eyebrows at lower resolution can look like closed eyes and closed eyes can almost disappear; in this case the eyebrows usually become the final result of detection but the coarse face region is still detected correctly. The face regions detected at coarse resolution determine a region of interest (ROI) in the original resolution, where the same detection algorithm is applied. Some results from the lower resolution, such as approximate mouth position, may be also used at fine resolution. The computational time is proportional to the ROI size, which is usually smaller than the size of the original image.

4 Experimental Results

Experiments were performed with our internal face database. The database contains frontal face images of 1,445 different people (10 images for each person) captured in an office, in total 14,450 color images of 720x480 pixels in size. The distance between the eyes varies from 50 to 100 pixels with average value equal to 73 pixels. Thus the scale factor of the face region varies in [0.68,1.36] interval. The following sizes of the facial feature templates were selected: 54x40 pixels for the templates in Fig.3a,b, 36x12 pixels for the templates in Fig.3c,d. The entire database was processed with the same set of algorithm parameters.

For accuracy evaluation, the ground truth (GT) eye positions were marked manually in 2890 images (2 images for each person. Table 1 shows reliability of detection expressed as: Precision defined as the ratio Number of detections / (Number of detections + False alarms) and Recall defined as the ratio Number of detections / (Number of detections + Number of missed detections). The Central displacement magnitude (defined as the distance between the center of the GT-eyes and the center of the detected eyes), and Relative central displacement (defined as the ratio of the central displacement magnitude to the distance between the GT-eyes) show the location accuracy. Fig.6 shows distribution of coordinate difference between the detected eye position, denoted as (STM_X,STM_Y), and GT positions. These Gaussian-like distributions show that in most cases the detected eye positions are close to GT and the vertical coordinate is measured more accurately than the horizontal coordinate.

Another series of tests was performed with images compressed with different JPEG quality levels. For JPEG compression we used standard "Baseline Sequential" option.

Precision	0.95
Recall	0.97
Mean central displacement, pixels	2.67
Relative central displacement	0.04

Table 1. Accuracy measurements

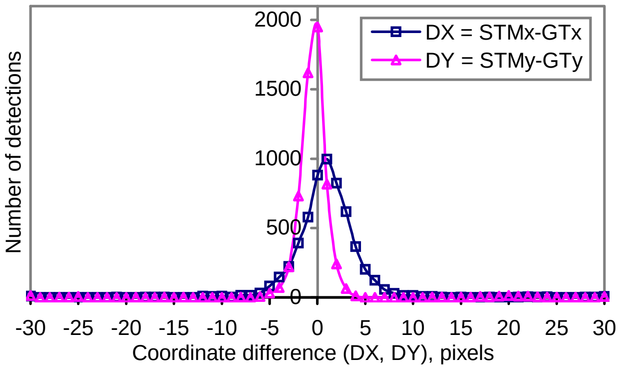

Fig.6 Accuracy of the eye positions. Both left and right eye contributed to the distributions independently.

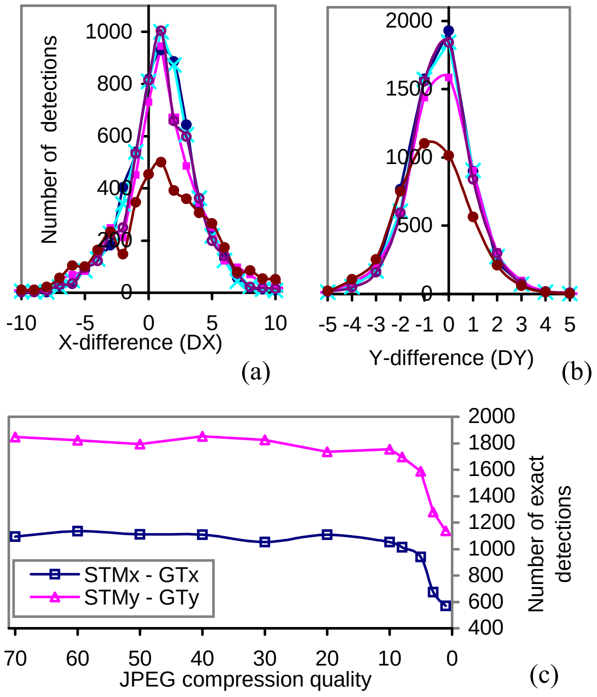

Fig.7 Method robustness to image compression.

Even at the lowest quality of compression the main face features are still recognizable by human and they do not loose their similarity to STM templates (Fig.3). The compression process produces blocking artefacts, which lead to the appearance of noisy edges, loss of colour information and distortion of facial features. Noisy edges do not much affect the detection results, because our method uses only first and second order statistics of large areas. The gradient features (7),(8) are computed only in small and narrow eye and mouth regions, and they are more or less preserved after compression. Loss of colour information does not affect the detection results because only greyscale information is used. Distortion of facial features caused by the compression can affect some detection algorithms that use local image features, but this is not the case with the STM-based method, which uses only relatively large generic features. Fig.7 shows how JPEG compression impacts on the detection accuracy. Each image from the GT-subset of the database was compressed with quality levels varying from 70 to 1 and for each level the location error distribution similar to the one shown in Fig.6 was created. Some of these distributions are shown in Fig.7a,b (separately for X- and Y-coordinates). The curves with the highest peak correspond to quality level=70; the curves with the lowest peak correspond to the quality=1. Then the values of each peak were used to create the graphs in Fig.7c. These graphs show that only the lowest compression quality values (below 10) greatly influence the detection results.

As we have shown, the method can accurately detect eye positions in large variety of conditions (different face sizes and image quality). The requirement of real-time performance often conflicts with the requirements of robustness and accuracy, but in our implementation we achieved good performance results (Table 2) even without assembler-based optimization. The testing was performed on a Pentium IV, 3GHz processor. The detection algorithm, described in the Section 3, takes about 9ms for a standard VGA image size, thus leaving more than 20ms for other image processing tasks in a real-time system working at 30 frames per second rate.

Finally, we show some results of our method using the BioID face database [6]. Fig.8 shows typical detection results. Fig.9 shows an experiment with image compression. In this particular case the method works with all compression quality levels.

5 Conclusion

A robust, accurate and high performance object detection method has been presented. The method can handle moderate scale factors of the object and is robust to local image warping. The latter was demonstrated by detection of such highly variable objects as faces and facial features. Successful object detection tests were performed in JPEG images having compression quality as low as 1-3 out of 100.

Test data	Image size, pix.	Average time per image, ms
Full Mitsubishi Electric database – 14450 images	720x480	9.4
2 min of web-camera video, 2254 frames	320x240	2.7

Table 2. Performance of the method

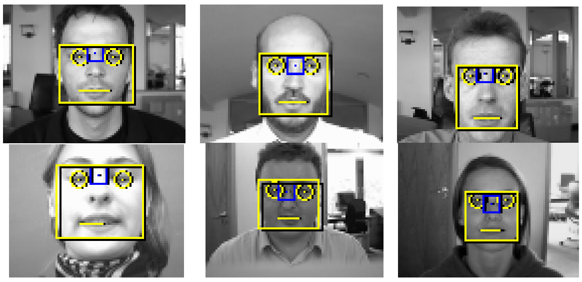

Fig.8 Facial feature detection examples in BioID face database

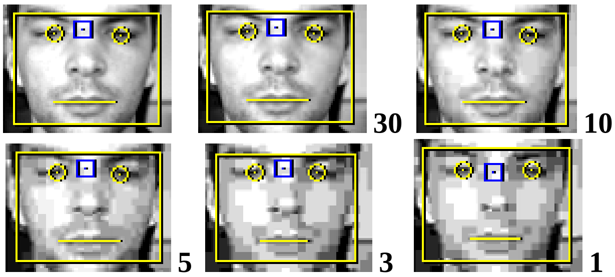

Fig.9 Method robustness to image compression. Top-left image is original quality image, which was JPEG-compressed using 30,10,5,3,1 quality levels.

References:

[1]N.Sebe, M.Lew, D.Hujismans, Toward Improved Ranking Metrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1132-1142, 22(10), 2000

[2]K.Chung, Fast Stereo Matching Using Rectangular Subregioning and 3D Maximum-Surface Techniques, *International Journal Of Computer Vision*, vol.47, no.1/2/3, pp.99-117, 2002

[3]A.Jain, Y.Zhong, S.Lakshmanan, Object Matching Using Deformable Templates, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, Issue 3, pp267-278, 1996

[4]Y.Keller, A.Averbuch, Unified Approach To FFT-Based Image Registration, *ICASSP*, Orlando, USA, May 2002

[5] http://www.uk.research.att.com/facedatabase.html

[6]R. Frischholz, U. Dieckmann, BioID: A Multimodal Biometric Identification System, In *IEEE Computer*, Vol. 33, No. 2, February 2000.