AUTOMATIC EXTRACTION OF FRONTAL FACIAL FEATURES

Vladimir Vezhnevets, Stanislav Soldatov, Anna Degtiareva

Dept. of Computational Mathematics and Cybernetics
Moscow State University
Moscow 119899, RUSSIAN FEDERATION
{vvp@graphics.cmc.msu.ru}

In Kyu Park

Multimedia Laboratory
Samsung Advanced Institute of Technology
Yongin 449-712, REPUBLIC OF KOREA
{pik@ieee.org}

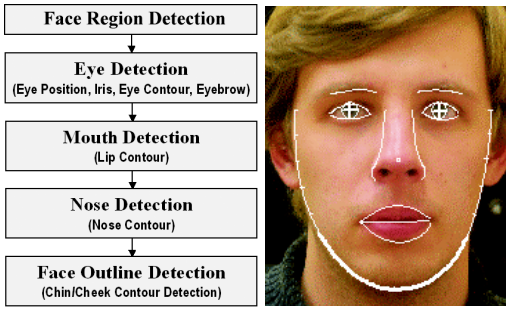
Abstract

This paper describes algorithms for face and facial features detection in still frontal images. It is designed for the task of automatic image-based 3-D face modelling. This requires the detection to be accurate enough to produce exact facial features and robust to images of widely varying quality and picture taking conditions. The facial feature detection task is solved in several steps. First, facial area is detected using a novel method based on skin color segmentation and adaptive ellipse fitting. Next, eye positions are estimated by finding eye-shaped and eye-sized areas of red channel sharp changes. Finally, exact facial contours of eyes, eyebrows, nose, mouth, chin, and cheek are estimated by employing deformable models, template matching, and color segmentation. The main contribution of this paper is a set of innovations in face and facial feature detection algorithms that achieve high detection robustness and accuracy.

1. INTRODUCTION

The detection of face and facial features has been receiving researchers' attention during past a few decades. The ultimate goal has been to develop algorithms equal in performance to human vision system. In addition, automatic analysis of human face images is required in many fields including surveillance and security, human computer interaction (HCI), object-based video coding, virtual and augmented reality, and automatic 3-D face modelling.

The algorithms for face and facial feature detection can be divided roughly into two broad categories. The techniques in the first category are based on human expertise and try to transform human knowledge and experience into formal algorithms [1][2][3][4][5]. The methods from the second category try to obtain the face knowledge implicitly from training images using the pattern recognition algorithms [1]. The main concern during development of our approach is to combine fast processing and detailed feature information of the first category techniques with robustness to wide class of facial images.

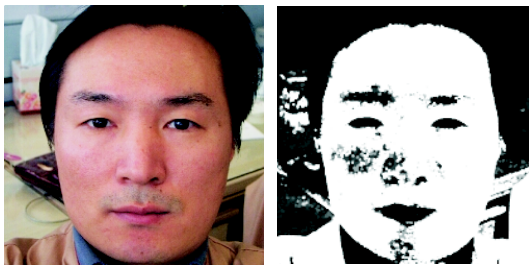


(a) (b)

Fig. 1. Algorithm overview and a typical result. (a) Block diagram of the proposed approach. (b) Feature detection result.

This paper describes models and algorithms for face and feature detection which are robust to different lighting, imaging conditions and quality. The proposed approach is divided into several steps as shown in Figure 1 (a). Initially, a skin color model is used to find pixels with color close to human skin. Then, a deformable elliptic model helps to find the face candidate region. Afterwards, eye positions are found by analyzing red image channel variations. After the eyes are found, the face region bounding box is updated to conform to reasonable human face proportions so that the image is cropped, rotated and scaled to create an upright face-only image of a fixed scale. Finally, facial contours (eyes, eyebrows, nose, mouth, chin and cheek) are detected by algorithms described in further sections. An example of the feature detection is shown in Figure 1 (b).

This paper is organized into several sections. In Section 2, we present the method for detecting face region. Section 3 describes the algorithm for detecting eye features including eye position, iris, eye and eyebrow contour. In Section 4 and Section 5, the methods for detecting lip and nose contours are proposed, respectively. Section 6 addresses the technique to find the contours of chin and cheek. Experimental results are shown in Section 7. Finally, we give a conclusive remark in Section 8.



(a) (b)

Fig. 2. Typical skin detection examples. (a) Initial image. (b) Image with skin pixels marked as white.

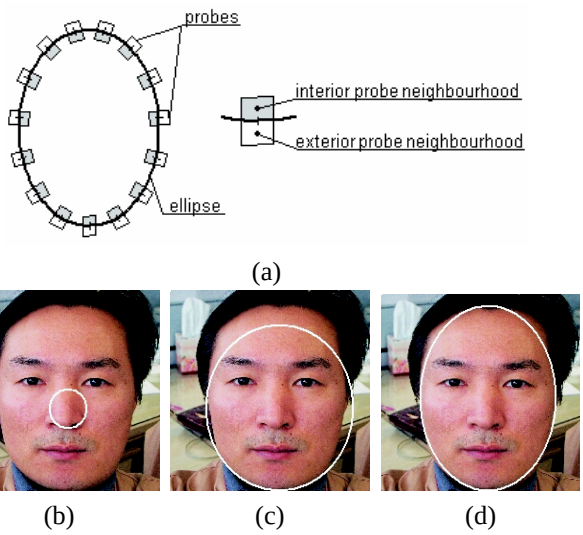
2. FACE REGION DETECTION

Many face detection algorithms have been proposed, exploiting different heuristic and appearance-based strategies (a comprehensive review is presented in [1]). Among those, color-based face region detection has gained strong popularity, for it enables fast localization of potential facial regions and is highly robust to geometric variation of face patterns and illumination conditions (except colored lighting). The success of color based face detection depends heavily on the accuracy of the skin color model. We employ Bayes skin probability map (SPM) [6] in normalized $R-G$ chrominance color space, that has shown good performance for skin color modelling. Typical skin detection results are presented in Figure 2.

Skin color alone is usually not enough to detect potential face regions reliably due to possible inaccuracy of camera color reproduction and presence of non-face skin-colored objects. Popular methods for skin-colored face region localization are based on connected components analysis [2] and integral projections [7]. Unfortunately, these simple techniques fail to segment facial region reliably in cases when face is not well separated from what is mistakenly classified as 'skin', as shown in Figure 2 (b). To cope with these problems a deformable elliptic model shown in Figure 3 (a) was developed. The model is initialized near the expected face position. For example, the mass center of the largest skin-colored connected region would be a nice choice as shown in Figure 3 (b). Then, a number of rectangular probes (we used twelve) on the ellipse border deform the model to extract an elliptic region of skin-colored pixels, as shown in Figure 3 (b) and (c). The densities of skin-colored pixels in probe's exterior and interior neighborhood control the probe displacement vector $\overrightarrow{v_i}$:

$$\overrightarrow{v_i} = \begin{cases} -k_{in} \cdot \overrightarrow{n_i}, \text{ if } \frac{N_{in}}{S_{probe}} < T_1, \\ k_{out} \cdot \overrightarrow{n_i}, \text{ if } \frac{N_{out}}{S_{probe}} > T_2, \\ 0, \text{ otherwise.} \end{cases} \quad (1)$$

In (1), i is the probe index, $\overrightarrow{n_i}$ is the probe expansion direction (normal to the model border, pointing outside ellipse), N_{in} and N_{out} are the numbers of skin pixels in the probe's interior and exterior neighborhood respectively, S_{probe} is the probe neighborhood area, and T_1, T_2 are threshold values. The k_{in} and k_{out} coefficients (≥ 0) control the model deformation behavior.



(a)

(b) (c) (d)

Fig. 3. Face region detection with deformable model. (a) Deformable model. (b) Convergence of the deformable model.

After displacement vectors for all probes are calculated, an ellipse is fitted to the centers of the repositioned probes. The ratio of major/minor axes length for the expected face ellipse is restricted to be less than 1.5. Taking advantage of the expected shape of skin-colored region (ellipse) makes the method more robust, compared to existing methods for skin pixels cluster detection (i.e. [8]).

3. EYE FEATURE DETECTION

3.1. Eye Position Detection

Accurate eye detection is very important in the subsequent feature extraction, since the eyes provide base information about expected location of other facial features. Most eye detection methods exploit the observation that eye regions usually differ significantly from the surrounding skin and exhibit sharp changes in both luminance and chrominance. Those methods employ integral projections [9], morphological filters [7], edge map analysis [3], non-skin color areas [2] [10] detection and other techniques to find potential eye locations in the facial image. The practice shows that using non-skin color and low pixel intensities as eye region characteristics can lead to severe detection errors. Luminance edge analysis needs a good facial edge map, which is hard to obtain when the image is noisy or low contrasted. Detection

Fig. 4. Results of eye position detection.

of sharp changes in red image channel gives more stable result, because iris usually exhibits low values of red (both for dark and light eyes), compared to surrounding pixels (eye white and skin). We propose a method that can effectively detect eye-shaped variations of the red channel, while being easily implemented through composition of convolution and basic arithmetic operations on images.

To detect the intensity change more easily, the image red channel intensities are stretched to the maximum range and a variation image is calculated by the formula given by

$$V_n(x,y) = \frac{\alpha}{|R_{n,x,y}|} \sum_{r \in R_{n,x,y}} \left\{ I(r) - \frac{1}{|P_{n,r}|} \sum_{p \in P_{n,r}} I(p) \right\}^2 \quad (2)$$

Here, I is the original red channel image, p and r denote pixel locations, $R_{n,x,y}$ is a rectangle of $(n \times 7)$ size, centered at (x, y) and $P_{n,r}$ is an ellipse of $(n \times n/3)$ size, centered at r. The variation image calculation parameters are: the scaling coefficient, α, and the expected size of the eye features, n. The meaning of the variation image $V_n(x, y)$ can be described as the dilatation of the high-frequency patterns in the red channel facial image. The variation image is calculated for several values of n with carefully chosen α's to cope with high variance of the eye appearance. This results in stable and correct behavior for images of different lighting and quality. Next, the connected components of pixels with high variation values are tested to satisfy shape and size restrictions and positioned symmetrically to get the best-matching eyes positions.

3.2. Eye Contours Detection

Our eye contour model consists of upper lid curve (in cubic polynomial), lower lid curve (in quadratic polynomial) and the iris circle. The iris center and radius are estimated with the algorithm developed by Ahlberg [11]. It is based on the assumptions that the iris is approximately circular and it is dark against the background, *i.e.*, the eye white.

Conventional approaches of eyelid contour detection use deformable contour models attracted by high values of spatial luminance gradient [5]. Deformable models need careful formulation of the energy term and good initialization, otherwise unexpected contour extraction result can be acquired. Moreover, it is undesirable to use luminance edges for the contour detection, because eye area may have many outlier edges. We propose a novel robust technique, that achieves stability and accuracy. Taking luminance values along a single horizontal row of an eye image as a scalar function $L_y(x)$, it is easily observed that the significant local minima correspond to eye boundary points, as shown in Figure 5. This observation holds valid for many eye images taken under very different lighting conditions and quality.

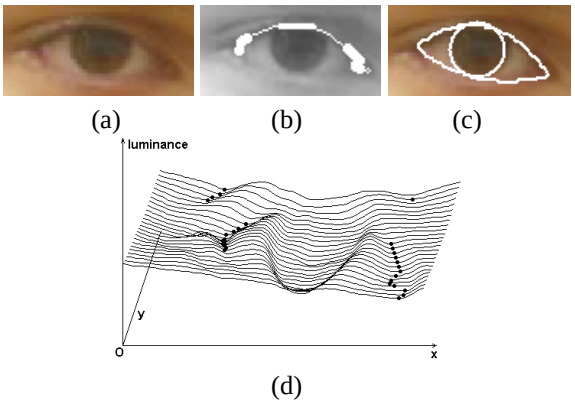


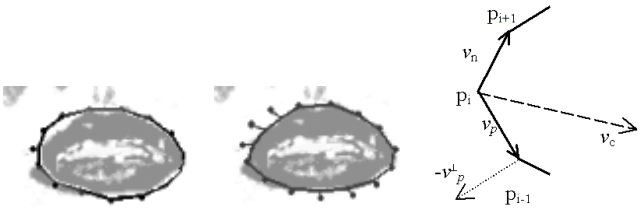
Fig. 5. Eye contours detection. (a) Source image. (b) Detected eyelid points and fitted curve. (c) Detected contour. (d) Pseudo 3-D luminance graph of the eye area. Probable eye border points marked with dark circles.

Detected candidate pixels of eye boundary are filtered to remove outliers, before fitting a curve to upper eyelid points. On the other hand, the lower lid is detected by fitting the eye corners and the lower point of the iris circle with a quadratic curve. The detailed description of our method can be found in [12].

Eyebrows are detected by fitting parabolas to the dark pixels after binarizing the luminance image in the areas above the eye bounding boxes.

4. LIP CONTOUR DETECTION

In most cases, lip color differs significantly from that of the skin. We use an iteratively refined skin and lip color models to discriminate lip pixels from the surrounding skin. The pixels classified as skin at the face detection stage and located inside the face ellipse are used to build person-specific skin color histogram. The pixels with low values of person-specific skin color histogram, located in the lower face part are used to estimate the mouth rectangle. Then, skin and lip color classes are modelled by two-dimensional Gaussian probability density functions in $(R/G, B/G)$ color space.



(a) (b) (c)

Fig. 6. Lip contour detection. (a) First step of contour fitting. (b) 12th step of contour fitting. (c) Layout of control forces.

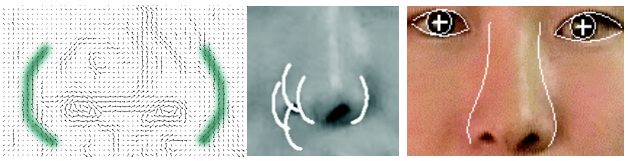
It is observed empirically that this color space shows excellence in discriminating lip from skin. The difference between lip and skin probability for each pixel is used to construct so-called 'lip function image' which is shown in Figure 6 (a) and (b). First, the initial mouth contour is roughly approximated by an ellipse which is computed from the moments of inertia of pixels with high lip function values. The contour points move radially outwards or inwards, depending on lip function values of pixels they encounter. Three forces control the i_{th} contour point's displacement, as given by

$$F_i = F_i^{data} + F_i^{form} + F_i^{sm} \tag{3}$$

$$\text{where} \begin{cases} F_i^{data} = \begin{cases} -k_{out}^i \nu_c, f(p_i) \geq T \\ k_{in}^i \nu_c, f(p_i) < T \end{cases} \\ F_i^{form} = -k_{form} \frac{(\nu_p + \nu_n) \cdot \nu_c}{|\nu_p + \nu_n|} \\ F_i^{sm} = -2k_{sm}(1 + \nu_p \cdot \nu_n) \\ \qquad sgn((p_{i+1} - p_{i-1}) \cdot (-\nu_p^{\perp})) \end{cases} \tag{4}$$

where k_{in}, k_{out}, k_{form}, and k_{sm} are positive coefficients that control the contour deformation behavior; $f(p_i)$ is the value of lip function in point p_i; T is a threshold; ν_n and ν_p are unit vectors pointing to next and previous contour points; ν_c is a unit vector, pointing to the ellipse center; $-\nu_p^{\perp}$ is a unit vector which is rotated by $\frac{\pi}{2}$ counterclockwise from $-\nu_p$. In Figure 6 (c), the layout of the vectors involved is illustrated.

In (3) and (4), F_i^{data} is the data force which controls the growth and contraction of the contour. F_i^{form} keeps the contour shape close to an ellipse. To make this force affect global contour form, ν_n and ν_p are taken as average of several neighbor points. It reaches its minimum, when $\nu_n + \nu_p$ and ν_c are aligned in same direction and reaches its maximum when they are perpendicular. F_i^{sm} controls the smoothness of the contour, while constraining each point not to go outside. We set the force direction to be parallel with the ellipse radii $-\nu_c$, which results in more stable behavior during convergence. The mouth contour points are moved until their displacements become less than a given threshold. Note that 30 iterations is enough in most cases.



(a) (b) (c)

Fig. 7. Nose contour detection. (a) Luminance gradient vector field of the nose area, with marked nose side parts. (b) Nose sides candidates. (c) Detected nose curves.

To match the lip corners exactly, the smoothing force F_i^{sm} is weakened for the points expected to become the lip corners, by lowering k_{sm} for the corner neighbor points. Lower part of mouth contour should be more smoothing than upper part, which is also considered in assigning k_{sm}. The process of contour fitting is illustrated in Figure 6 (b) and (c).

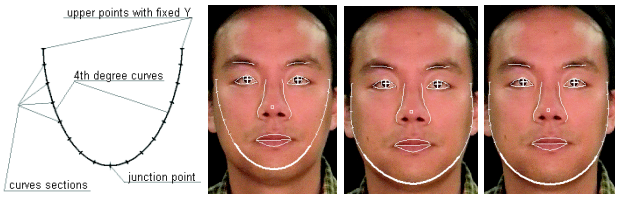
5. NOSE CONTOUR DETECTION

The representative shape of nose sides has already been exploited for the sake of increasing the robustness. Matching of nose side template to the edge and dark pixel pattern has been successful in [4] and [5]. However, in cases of blurry picture or ambient face illumination, it becomes more difficult to utilize the nose edge patterns. In such cases, edge magnitude and pixel brightness information are not enough to detect reliable features. In our approach, we make a step further from the naive template methods. The proposed approach utilizes full information of the gradient vector field (both magnitude and direction). Our nose side template represents typical nose side shape in a fixed scale. To compensate the fixed scale of the template, the nose image (face part between the eye-centers, vertically from eyes bottom to top of mouth box) is cropped and scaled to fixed size. Median filtering is applied before template matching to alleviate the noise, while preserving the edge information. Luminance gradient vector field is calculated with Prewitt edge masks, which apply less smoothing than Sobel masks, when calculating the luminance derivatives. The result of luminance gradient vector field is shown in Figure 7 (a). The figure-of-merit (FoM) at a pixel location q is defined by the following formula

$$FoM(q) = \sum_{p \in S(q)} Fit(p) \tag{5}$$

$$Fit(p) = \begin{cases} 1, \text{if } \exists r \in \Omega(p) \text{s.t. } \|\overrightarrow{\nabla I(r)} \cdot \overrightarrow{a(p)}\| \geq 0.9 \\ \quad \text{and } |\overrightarrow{\nabla I(r)}| > T_1 \\ 0, \text{otherwise.} \end{cases} \tag{6}$$

where, p, q and r denote pixel locations; $S(q)$ is the set of template curve points; $\Omega(p)$ is a 5x5 neighborhood of



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

Fig. 8. Chin and cheek contour detection. (a) Deformable model. (b)(c)(d) Model and initialization and fitting.

p; $\nabla\overrightarrow{I(r)}$ is the image spatial luminance gradient vector at r; and $\overrightarrow{a(p)}$ is the template curve tangent at p. T_1 sets the minimum gradient magnitude value to exclude weak edges. $Fit(p)$ is the counter function that yields the number of pixels in $\Omega(p)$ which exhibit significant gradient magnitude with gradient direction close to template curve tangent. Approximately 13% of maximum figure-of-merit template positions form a set of nose sides candidates as shown in Figure 7 (b). A pair of candidates, located with the close vertical coordinates on an approximately even distance from the face central line is chosen as the most probable nose position as shown in Figure 7 (c). As shown in Figure 7 (c), the final nose curves are estimated from the nose sides position and considering geometric relation on general human face structure about nose position.

6. CHIN AND CHEEK CONTOUR DETECTION

Deformable models have been proved to be an efficient tool for chin and cheek contour detection [13]. However, the edge map, which is the main information source for the face boundary estimation, results in very noisy and incomplete face contour information in some cases. A subtle model deformation rule, derived from general knowledge on human face structure must be applied for accurate detection [13]. We propose a simpler but robust method that relies on a deformable model, consisting of two fourth degree polynomial curves, linked at the bottom point as shown in Figure 8 (a). The model deformation process is designed in such a way that permits detection of exact facial boundary in case of noisy or incomplete face contour information provided by the edge map. Gradient magnitude as well as gradient direction is utilized together.

The model initial position is estimated from the already detected facial features as shown in Figure 8 (b). After the initialization, the model starts expanding towards the face boundaries, until it encounters strong luminance edges which are collinear with model curves. The model curves are divided into several sections. It expands until sufficient number of pixels occupied by a curve section has edge magnitude higher than a given threshold and edge direction collinear with model curve. After the evaluation of each section's displacement the model curves are fitted to the repositioned sections points by least squares fitting. This is repeated until the model achieves stable configuration. Figure 8 (b)$\sim$(d) illustrates this process, showing the convergence to the actual chin and cheek boundary. The lower chin area may exhibit weak edges or no edges at all. In this case, the lower model curve sections stop movement when they reach significant luminance valleys. In order to constrain the model expanding downwards excessively, the lower model point is not allowed to move lower than a level which is derived from the human face proportions.

7. EXPERIMENTAL RESULTS AND DISCUSSION

In order to evaluate the performance of the proposed algorithms, we tested about 30 photos of different people with different races (European and Asian), image resolutions (from 1000×1000 to 300×300), illumination conditions (front directional and ambient), and genders. Algorithms have shown good robustness and reasonable accuracy for the photos from our test set, compared with human feature detection as shown in Figure 9.

The whole facial feature set is detected fully automatically and has been used along with other information as input data for a 3-D image-based human head modelling system. The overall feature extraction time with the described algorithms is about 15$\sim$20 seconds per one photo on a Pentium IV 1.8GHz system. The most time consuming operation is the nose shape detection, because it performs search over all possible nose template locations. The processing time may be seriously reduced by algorithms and their implementation optimization, which have not yet performed.

8. CONCLUSION

A novel and practical facial feature extraction system has been described, which combines good accuracy of feature detection with robustness to images of different quality and picture taking conditions. The procedure is fully automatic, while user modification is also allowed if necessary.

The detection results are used in the system for automatic 3-D image-based head modelling, and help it to achieve satisfactory face models. Also, the results can play a crucial role in feature-based face recognition system.

9. REFERENCES

[1] M. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, January 2002.

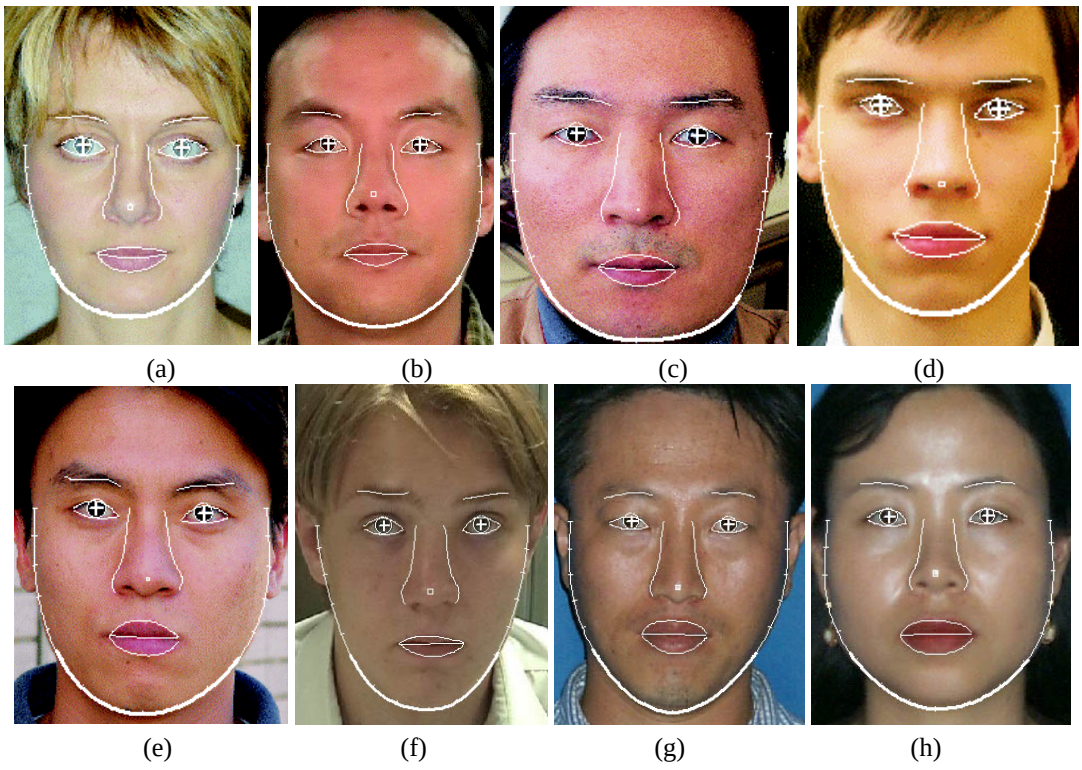


Fig. 9. Results of final feature detection.

[2] A. Yilmaz and M. Shah, "Automatic feature detection and pose recovery for faces," in *Proc. Fifth Asian Conference on Computer Vision*, January 2002, pp. 284–289.

[3] X. Zhu, J. Fan, and A. Elmagarmid, "Towards facial feature extraction and verification for omni-face," in *Proc. IEEE International Conference on Image Processing*, September 2002, vol. 2, pp. 113–116.

[4] Lijun Yin and Anup Basu, "Nose shape estimation and tracking for model-based coding," in *Proc. IEEE International Conference on Acoustics, Speech, Signal Processing*, May 2001, pp. 1477–1480.

[5] M. Kampmann and L. Zhang, "Estimation of eye, eyebrows and nose features in videophone sequences," in *Proc. International Workshop on Very Low Bitrate Video Coding*, October 1998, pp. 101–104.

[6] V. Vezhnevets, V. Sazonov, and A. Andreeva, "A survey on pixel-based skin color detection techniques," in *Proc. Graphicon-2003*, September 2003.

[7] L. Jordao, M. Perrone, J. Costeira, and J. Santos-Victor, "Active face and feature tracking," in *Proc. International Conference on Image Analysis and Processing*, September 1999, pp. 572–576.

[8] K. Toyama, "Look, ma - no hands!' handsfree cursor control with real-time 3d face tracking," in *Proc. Workshop on Perceptual User Interfaces*, November 1998, pp. 49–54.

[9] S. Baskan, M. Mete Bulut, and Volkan Atalay, "Projection based method for segmentation of human face and its evaluation," *Pattern Recognition Letters*, vol. 23, no. 14, pp. 1623–1629, 2002.

[10] Gu. C. Feng and P. C. Yuen, "Multi-cues eye detection on gray intensity image," *Pattern Recognition*, vol. 34, no. 5, pp. 1033–1046, May 2001.

[11] Jorgen Ahlberg, "A system for face localization and facial feature extraction," Tech. Rep. LiTH-ISY-R-2172, Linkoping University, 1999.

[12] V. Vezhnevets and A. Degtiareva, "Robust and accurate eye contour extraction," in *Proc. Graphicon-2003*, September 2003.

[13] M. Kampmann, "Estimation of the chin and cheek contours for precise face model adaptation," in *Proc. International Conference on Image Processing*, October 1997, vol. 3, pp. 300–303.