Parametric models for facial features segmentation

Zakia Hammal, Nicolas Eveno, Alice Caplier, Pierre-Yves Coulon

Parametric models for facial features segmentation.

Z. HAMMAL, N.EVENO, A.CAPLIER, PY.COULON

LIS, INPG, 46 avenue Félix Viallet, 38031 Grenoble Cedex, FRANCE

alice.caplier@lis.inpg.fr

Abstract – In this paper, we are dealing with the problem of facial features segmentation (mouth, eyes and eyebrows). A specific parametric model is defined for each deformable feature, each model being able to take into account all the possible deformations. In order to initialize each model, some characteristic points are extracted on each image to be processed (for example, eyes corners, mouth corners and brows corners). In order to fit the model with the contours to be extracted, a gradient flow (of luminance or chrominance) through the estimated contour is maximized because at each point of the searched contour, the gradient (of luminance or chrominance) is normal. The definition of a model associated to each feature offers the possibility to introduce a regularisation constraint. However, the chosen models are flexible enough to produce realistic contours for the mouth, the eyes and the eyebrows. This facial features segmentation is the first step of a set of multi-media applications.

Keywords: parametric models, deformable model, facial features, segmentation

1. Introduction

Facial features deformations contribute to the communication between humans: for example, lip reading allows to improve the understanding of a noisy vocal message and it is also a support of communication with hard of hearing people.

The aim of our work is the extraction of the contours of permanent facial features (mouth, eyes and eyebrows) with enough accuracy to be able to improve the human-to-human communication through a machine.

Several applications are under consideration:

- The outer contour of lips can be used in a mobile phone application in order to improve the vocal message quality in case of noisy transmission;
- This contour is going to be used in a project of phone device for hard of hearing people. This requires speech synthesis from lip shape and motion;
- The detected iris contour is going to be used to evaluate the vigilance or interest level of a user by the analysis of his frequency blinking. It will also be used to estimate the gaze direction of a PC user;
- All the detected contours (eyes, brows and lips) are used in an automatic system of facial emotion recognition based on video data.

All the considered applications require very accurate contours and real time processing.

In this paper, we propose new parametric models well suited for eyes, mouth and brows segmentation. For the initialisation of each chosen model, some characteristic points (eyes and brows corners for example) are extracted. Initial models are then deformed in order to maximize a gradient flow (of luminance and/or chrominance). The originality of our approach is the definition of parametric models able to take into account all the possible deformations of each considered feature. This yields to realistic and accurate segmentation.

Many algorithms have been proposed to solve the problem of lip segmentation. Some methods use low-level spatial information such as colour [29]. No smoothing constraint or shape constraint is considered so that the extracted contours are often of coarse quality. In [18], a *linear discriminant analysis* (LDA) is used to segment lip pixels from skin pixels. This yields to lip contours. Though the discriminant analysis is followed by a smoothing step, the segmentation remains noisy. *Snakes* [16] have been widely used for lip segmentation ([1][20]) because snakes can take into account in a same framework smoothing and elasticity constraints. Snake-based methods yield to interesting results but the main drawback is the tuning of several parameters. Moreover, the quality of the segmentation is dependent of initialization. A priori shape models can be used to obtain smoother contours: the extracted contour belongs to the possible space of the lip shape. For example, Active Shape Models (ASM) have been used [6]. But this approach requires a very huge learning database in order to be able to take into account all the possible deformations of a speaking mouth. And each image of the learning database has to be calibrated with very high accuracy: face orientation and illumination conditions have to be controlled. One possible solution to avoid learning stage is the use of parametric models [28].

Segmentation of eyes and brows is simpler because such features are less deformable than the mouth. Two kinds of approach exist. First of all a coarse localization of these features based on luminance information is extracted (valley images for example) ([19], [25]). These approaches yield to a very few accurate contours detection for eyes and brows. The

second approach introduces models to be related to the searched contours [22].

In our approach, parametric models are considered. In section 2, chosen parametric models for mouth, eyes and brows are described and justified. In the pre-processing step described in section 3, face illumination variations are removed and iris circular contour is extracted. Section 4 presents the algorithms for the automatic extraction of facial characteristic points used for the initialization of the parametric models. Section 5 describes the deformation model process according to gradient information to make the model coincide with the contours of the processed image. Section 6 gives qualitative and quantitative results in order to evaluated the pertinence of the chosen parametric models and the quality of the segmentation. The accuracy of the extracted contours is compliant with an application of facial emotion recognition.

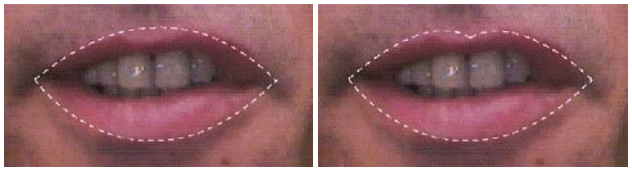
2. Model choice

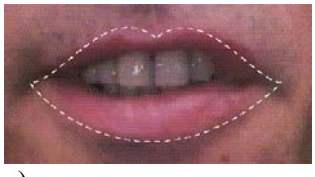
The analysis of face images coming from different databases shows that the models that have been proposed until now for mouth, eyes and brows are too rigid to obtain realistic contours.

2.1 Mouth model

Several parametric models have been proposed to model the lip contour. Tian [21] uses a model made of parabolas. This is very simple but the precision is limited (see Figure 1-a). Others authors propose to model the upper lip contour with 2 parabolas [5] or to use quartics [13]. It improves accuracy, but the model is still limited by its rigidity, particularly in the case of asymmetric mouth shape (see Figure 1-a, Figure 1-b, Figure 1-c)

a) b)





c)

Figure 1: a) model with 2 parabolas; b) model with 3 parabolas; c) model with quartics.

The choice of the right model for lip is a great challenge because lip contour is highly deformable. The use of an a priori model in the segmentation step yields to a smooth contour. But if the chosen model is not suitable, segmentation result will be of bad quality.

We propose a model made of 5 independent curves. Each curve describes a part of the lips boundary. Between Q_2 and Q_4, the Cupidon's bow is drawn with a broken line and the other parts of the contour are approximated by 4 cubic polynomial curves γ_i (see Figure 2). We also consider that each cubic has a null derivative at key points Q_2, Q_4 or Q_6. For example, γ_1 has a null derivative at Q_2.

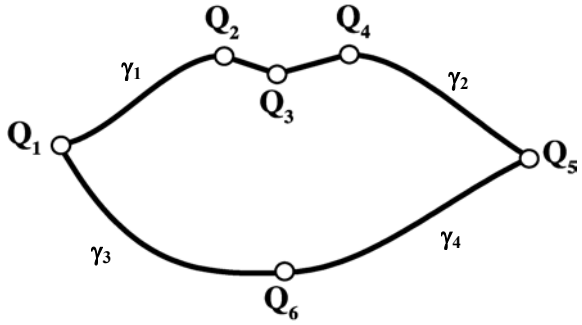
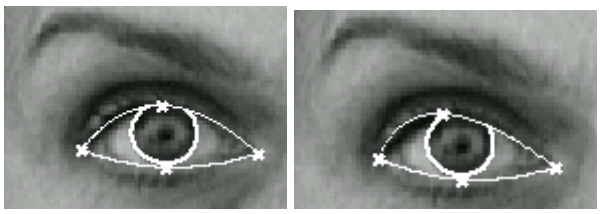


Figure 2: mouth parametric model

2.2 Eyes and brows parametric models

The most usual model for eyes is made of parabolas for eyelids and of a circle for iris ([5], [22]). But the analysis of multiples frames of eyes (on the ORL database [11]) shows that the contour of the upper eyelid does not always present a vertical symmetry. Figure 3 exhibits an example of non-symmetric eyelid that is not well approximated with a parabola (Figure 3-a) but the contour is well fitted with a Bezier curve (Figure 3-b). On the contrary, lower lip contour presents a vertical symmetry so that a parabola is well suited.



a) b)

Figure 3: a) model with 2 parabolas; b) model with a Bezier curve for the upper eyelid and a parabola for the lower one.

Regarding the eyebrows, they are often approximated by two broken lines defined by both corners and a middle point. This is a basic model. We propose the definition of a Bezier curve for each brow.

To summarize, we propose the following parametric model well suited to the multiple possible shapes of eyes and eyebrows (see Figure 4):

- For each eye: a circle for the iris (it could be a semi-circle if the eye is slightly closed); for the lower contour, a parabola defined by 3 points $\{P_1, P_2, P_4\}$; for the upper contour, a Bezier curve defined with 3 control points $\{P_1, P_2, P_3\}$; in case of closed eye, the considered model is a line defined by P_1 and P_2.
- For each brow: a Bezier curve defined with 3 control points $\{P_5, P_6, P_7\}$ for the lower contour (only the lower contour is extracted).

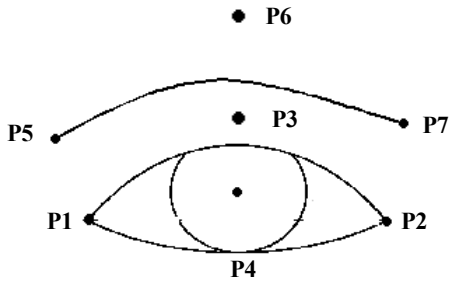


Figure 4: parametric model for eye and brow

The parametric model for eye and brow is less flexible than the model for the mouth because eyes and brows are less deformable. In each case, the aim is to propose the simplest well-suited model in order to reduce the complexity. For that reason, no link has been introduced between the relative positions of iris and eyes contours. This kind of link could be considered in a second step of temporal tracking of segmentation.

3. Pre-processing

3.1 Video acquisition conditions

Analyzed sequences are acquired with a Web-cam. Natural lighting conditions (white light) are considered (the aim is to do the acquisition with constraints compliant with the expected applications). The camera is facing the person (minimal admissible size for the face in the image is around 90x90 pixels). In the first frame, the face is supposed to be vertical and the eyes are supposed to be open. In the following frames, the face has to remain close to the vertical axis.

Several algorithms have been proposed for the automatic extraction of face in images (see the survey articles [14], [27]). Face extraction is beyond the scope of this paper and we use the MPT algorithm [30] based on the work of Viola and Jones [24]. This algorithm extracts a square bounding box around the face and gives two points located inside the irises. The facial square dimension is noted H_{face}. With this algorithm, pan and tilt head rotations are possible so long as facial permanent features remain visible. But the MPT algorithm is not very efficient in case of roll head motion: the detection of head that are not close to the vertical axis is not robust.

The case of a face with very long hairs, which are covering eyes and brows, is not considered: the facial features to be segmented are supposed to be visible.

3.2 Illumination variations attenuation

Illumination variations are attenuated by the use of a filter, which has the same behavior as the human retina [2]. This filter induces a local smoothing of illumination variations. Only the description of the filter is given here (for more details, see [2]). Retinal filtering induces a succession of filtering and adaptive compression. Let G be a Gaussian filter of size 15x15 and of standard deviation $\sigma = 2$. Let I_{in} be the initial image and let I_1 be the result of G filtering of I_{in}. Frame X_0 is defined by:

$$X_0 = \frac{0.1 + 410 I_1}{105.5 + I_1}$$

X_0 leads to the definition of the compression function C:

$$C : I \rightarrow \frac{(255 + X_0) I}{X_0 + I}$$

with I any frame.

Figure 5 gives the diagram of the retinal filter, I_{out} is the filtered image.

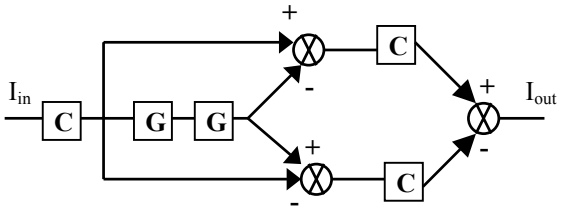


Figure 5: retinal filter

Figure 6 shows the result of retinal filtering on a face with lateral illumination. On that face, there is an important difference of lighting between right and left sides. After the filtering, illumination variations have been attenuated.

Figure 6: left, face with large illumination variations; right, filtered image.

3.3 Iris circle detection.

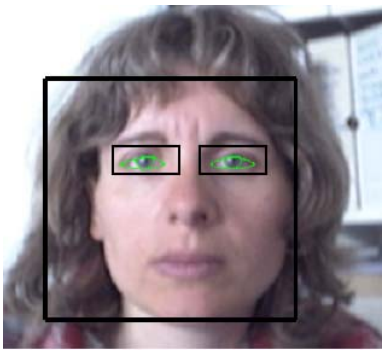


Figure 7: Results of face detection and eye bounding boxes construction.

Iris contour being the frontier between the dark area of iris and the eye white, it is supposed to be a circle made of points of maximum of luminance gradient. Since the eyes could be slightly closed, the upper part of the iris could be occluded. So for each iris, we are looking for the lower part of the iris circle.

With the face bounding box dimension H_{face}, a bounding box (H_{eye}, L_{eye}) around each eye has been defined with the following relations (see Figure 7):

$$H_{eye} = 0.1 * H_{face} \quad and \quad L_{eye} = 0.2 * H_{face}$$

These relations have been learnt on the 400 images of the ORL database [11]. Each bounding box is centred on the eye point given by the MPT algorithm. These relations are dependent on the algorithm used for face detection. If you change the face detector, a learning step is necessary in order to adapt the dimensions of the eye bounding box with respect to the dimension(s) of the face bounding box. Others face detectors can provide a rectangular bounding box and the limits of this bounding box with respect to the face can be different depending on the method used for the detection.

In each eye bounding box, each semi-circle of iris maximizes the normalized flow of luminance gradient (*NFLG*):

$$NFLG = \frac{1}{length\ SC} \sum_{p \in SC} \vec{\nabla} I(p).\vec{n}$$

where I_t is the luminance at point p and at time t, $\vec{n}(p)$ is the normal of the boundary at point p and SC is the lower semi-circle. The *length of the chosen contour normalizes the NFLG*. Several semi-circles scanning the search area of each iris are tested and the semi-circle, which maximizes the NFLG, is selected. Figure 8 shows that the right position of the semi-circle *SC* exhibits a sharp maximum of *NFLG* in the iris search area.

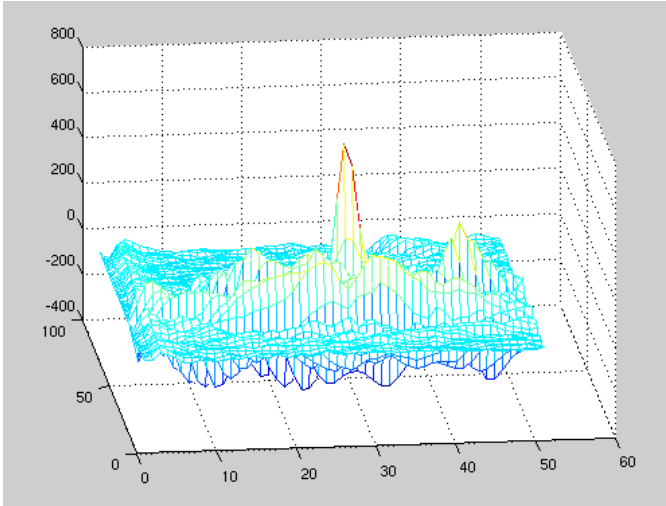


Figure 8: Evolution of the *NFLG* for each tested semi circle in the iris search area: *NFLG* is maximum when the selected semi circle coincides with the iris in the image.

Iris radius is supposed to be known and only the centre position of the searched semi circle is scanning the eye bounding box. The maximization of the *NFLG* is very fast (compared with a gradient flow slope method for example) and without any parameters tuning. It could be possible to solve the problem of the manual estimation of the iris radius in testing several radius values since iris radius value could be correlated to face dimensions.

4. Characteristic points extraction

Initial positioning of each model requires the automatic extraction of some characteristic points.

4.1 Eyes key points

A tracking process of points with maximum value of luminance gradient is used in order to localize eyes corners.

The centre of eye and brow bounding box coincides with the iris detected centre. This allows an efficient delimitation of search area for eye and brow.

Figure 9 gives an illustration of the corners detection method: starting with the points X_1 and X_2, pixels of maximal gradient of luminance located at two pixels from the vertical of the iris circle limits, a tracking process, to the left direction, of pixels with highest value of luminance gradient yields to the detection of the first corner C_1. Since the initial point X_1 is located below the corner C_1, only the three (black) points $\begin{pmatrix} \bullet & \bullet \\ \bullet & X_1 \end{pmatrix}$ located above and on the left of X_1 are tested. The curve between X_1 and C_1 (resp. X_2 and C_2) is made of pixels with local maximum of luminance

gradient. The tracking stops when the luminance gradient becomes negative since a skin pixel is clearer than an eye corner pixel (see Figure 9).

A similar tracking process to the right direction yields to the detection of the second corner C_2.

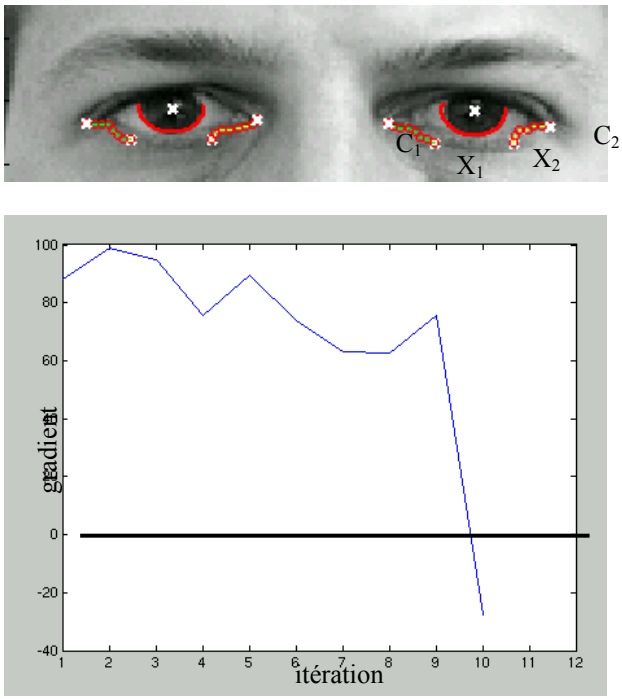


Figure 9: eye corners detection: top, tracking process; bottom, luminance gradient evolution along the X_1C_1 curve.

Points P_1 and P_2 of the model are fitted to both detected corners C_1 and C_2. Point P_4 of the parabola is aligned with the lowest point of the iris detected semi-circle and point P_3 initially coincides with the iris centre point. Figure 10 presents the result of the initialization of the eye models.

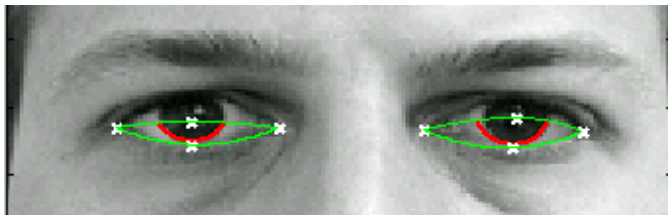


Figure 10: initialization of eye models

4.2 Eyebrows key points

Let S_1 and S_2 be the extremities of each eyebrow.

For each eyebrow, the search area is limited to an area located above the eyes. The abscissas x_1 and x_2 of both brow corners correspond to the left and right zero crossing of the derivative of the quantity (see Figure 11): $H(x) = \sum_{y=1}^{N_y} [255 - I(x,y)]$ and the ordinates y_1=y_2 correspond to the maximum of the quantity (see Figure 11):

$$V(y) = \sum_{x=1}^{N_x} [255 - I(x,y)]$$

where $I(x,y)$ is the luminance at pixel (x,y) and (N_x, N_y) represent the dimensions of the ROI for each eyebrow (ROI of the same sizes as the eye bounding box but located above the detected iris).

A third point S_3 is computed using S_1 and S_2 in the following way: $\{x_3 = (x_1 + x_2)/2; y_3 = y_1\}$.

The Bezier's curve for each eyebrow is initialized in taking $P_5 = S_1; P_7 = S_2; P_6 = S_3$.

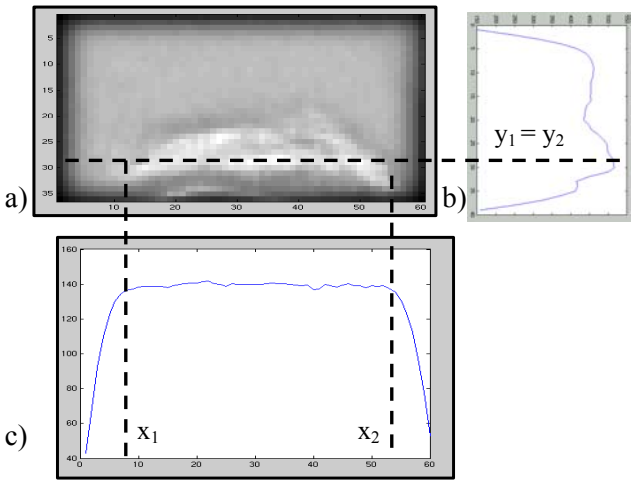


Figure 11: a) valley image; b) vertical projection V; c) horizontal projection H.

Brow corners detection is a difficult task especially for the external corner, which is not always well defined on the image to be processed. The proposed method supposes that brows are always visible and that brows are darker than the skin (which is generally the case). Since the results of brow corners detection method depend on the contrast of brows and on noise, initial detected brows corners will be adjusted during the deformation step of the brow model.

4.3 Mouth key points

The localization of characteristic points of the mouth is more complex. This detection uses mixed information of luminance and chrominance and a new kind of snake called « jumping snake ».

4.3.1 Skin and Lips Color Analysis

The choice of a color space is crucial because it has a direct influence on the robustness and the accuracy of the segmentation finally achieved. Note that

luminance does not contain enough information because of shadows that can occur on face (around the nose, below the inner lip for example). Our goal is to find a color space that enables a good separation of skin and lip pixels. In [29] the HSV system is used for the lips detection because hue has a good discriminative power. Even for different talkers, hues of lips and skin are relatively constant and well separated. But hue is generally very noisy because of its "wrap-around" nature (low values of hue lie close to high ones) and its very bad reliability for low saturation pixels.

CIELUV and YCrCb spaces can also be used in face analysis. It has been shown that the skin color subspace covers a small area of the (*Cr*,*Cb*) or (*u*,*v*) planes ([22][25]). However, the distributions of skin and lip colors often overlap and vary for different speakers. This makes these spaces unsuitable for lip segmentation.

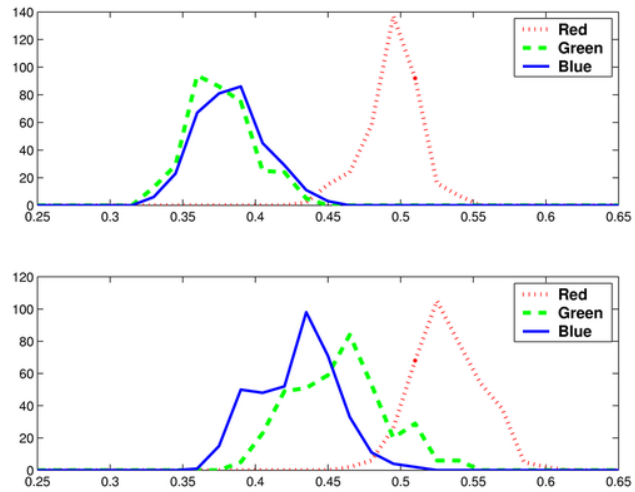


Figure 12: typical R (dots), G (dashed), B (plain) histograms of lip pixels (top) and skin pixels (bottom).

In *RGB* space, skin and lip pixels have quite different components. Figure 12 shows the histograms of each R, G, and B component for skin and lip respectively. For both, red is prevalent. Moreover there is more green than blue in the skin color mixture and for lips both components are almost the same [7]. Skin appears more yellow than lips because the difference between red and green is greater for lips than for skin. In order to distinguish skin pixels from lip pixels, the idea is to exhibit this difference, which can be done by the use of the pseudo-hue proposed in [15] and defined by:

$$h(x,y) = \frac{R(x,y)}{G(x,y)+R(x,y)}$$

where *R(x,y)* and *G(x,y)* are respectively the red and the green components of the pixel (x,y). Unlike usual hue, the pseudo hue is bijective. It is higher for lips than for skin [7] (cf. Figure 13).

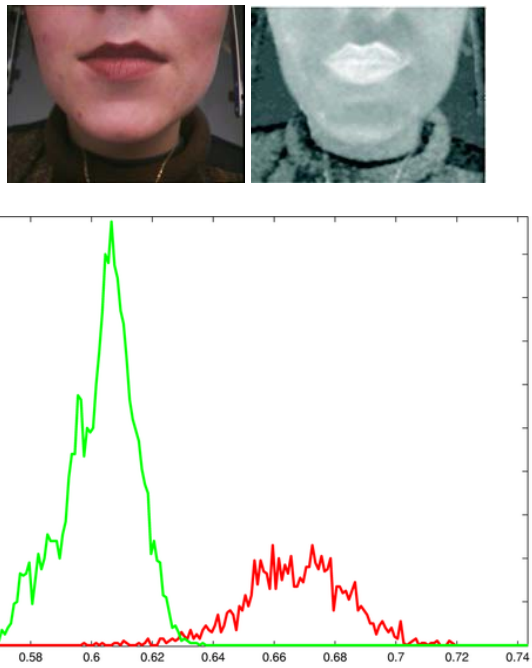


Figure 13: pseudo-hue image and histogram for lip in red (or dark) and for skin in green (or bright).

Luminance is also a good cue to be taken into account especially when the light comes from above the speaker In this case the top frontier of the upper lip is very well illuminated while the upper lip itself is in the shadow. This difference of illumination is not always totally removed by the preprocessing step.

To combine color and luminance information, we work in the RGB color space and we use the "hybrid edge" $R_{top}(x,y)$ introduced in [8]. It is computed as follows:

$$\vec{R}_{top}(x,y) = \vec{\nabla}\left[h_N(x,y) - I_N(x,y)\right]$$

where $h_N(x,y)$ and $I_N(x,y)$ are respectively the pseudo hue and the luminance of pixel (x,y), normalized between 0 and 1.

$\vec{\nabla}$ is the gradient operator. This hybrid edge exhibits much better the top frontier of the mouth than the classic gradients of luminance or pseudo-hue (see [7])

4.3.2 The Jumping Snake algorithm

Active contours, or snakes, have proved their efficiency in many segmentation problems. Since their introduction by Kaas et al. [16], many improvements have been proposed in the literature. But none of them has totally removed the two major weak points of the snakes: the choice of parameters and the high dependence on the initial position. The

method presented here helps addressing these problems.

To find the upper mouth boundary, we introduce a new kind of active contour that we call "jumping snake". Its convergence is a succession of jumps and growth phases [9]. It is initialized with a seed S^0 that can be located quite far away from the final edge (see Figure 14). The seed is put manually above the mouth and near its vertical symmetry axis. The snake grows from this seed until it reaches a pre-determined number of points. This growth phase is quite similar to the growing snake proposed by Berger and Mohr [3], in the sense that the snake is initialized with a single point and is progressively extended to its endpoints. Then, the seed "jumps" to a new position that is closer to the final edge. The process stops when the size of the jump is smaller than one pixel (which requires 5 iterations in average).

During the **growth phase**, left and right endpoints are added to the snake. They are located at a constant horizontal distance, denoted Δ, from the previous point. Moreover, the search area is restricted to the angular sector $(\theta_{\inf}, \theta_{\sup})$ (see Figure 15). The best left and right endpoints, denoted $M_{-(i+1)}$ and M_{i+1}, are found in this area by maximizing the $\vec{R}_{top}$ mean flow through the end segments $M_{-(i+1)}M_{-i}$ and M_iM_{i+1} (see Figure 15).

These two mean flows can be written as follows:

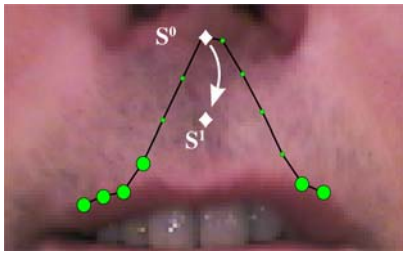


Figure 14: Jumping snake initialization and seed jump.

$$\phi_{i+1} = \frac{\int_{M_i}^{M_{i+1}} \vec{R}_{top}.d\vec{n}}{\left\|\overrightarrow{M_iM_{i+1}}\right\|} \qquad \phi_{-i-1} = \frac{\int_{M_{-i}}^{M_{-i-1}} \vec{R}_{top}.d\vec{n}}{\left\|\overrightarrow{M_{-i}M_{-i-1}}\right\|}$$

where $d\vec{n}$ is the vector orthogonal to the segment. The maximizations of ϕ_{-i-1} and ϕ_{i+1} are achieved by a systematic computation over a small set of candidates located in the search area. A method of gradient slope has been rejected since the number of points to be tested is low (around 8 points with $(\theta_{\inf}, \theta_{\sup}) = (-\frac{\pi}{3}, \frac{\pi}{5})$). The proposed optimization method being exhaustive, it allows to have an accurate solution very quickly.

When the jumping snake reaches a pre-determined number of points 2*N*+1, the growth stops and the position of the new seed S^1 is computed. This is the **jump phase** of the jumping snake algorithm. Let $\{M_{-N},...,M_{-1},S^0,M_1,...,M_N\}$ be the points of the snake and let $\{\phi_{-N},\ldots,\phi_{-1},\phi_1,\ldots,\phi_N\}$ be the mean flows through the 2*N* segments. The new seed S^1 has to get closer to high gradient regions, i.e. high mean flow segments. We consider that S^1 is the barycentre of S^0 and the points, which are in the highest gradient regions (the big dots on Figure 14). If $\{i_1,...,i_N\}$ are the indices associated to the *N* highest mean flows, then the vertical position of S^1 can be written as follows:

$$y_{S^1} = \frac{1}{2}\left(y_{S^0} + \frac{\sum_{k=1}^{N} \phi_{i_k} y(i_k)}{\sum_{k=1}^{N} \phi_{i_k}} \right)$$

where $y(i_k)$ is the vertical position of the point M_{i_k}. The horizontal position x_{S^1} of the seed is kept constant.

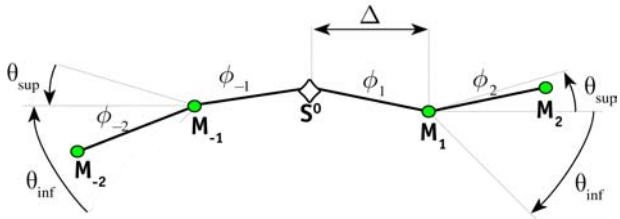


Figure 15: From the seed S_0, adding left and right endpoints extends the snake. The Rtop mean flows Φ_i through each segment have to be maximized.

Then, a new snake grows from this new seed until it reaches the predetermined length and "jumps" again. This growth jump process is repeated until the jumps amplitude becomes smaller than 1 pixel. Typically, 4 or 5 jumps are needed to achieve the jumping snake convergence.

The *jumping snake* algorithm involves 4 parameters: $(\theta_{\inf}, \theta_{\sup})$ for the definition of the angular sector of possible evolution for each segment of the snake; Δ the horizontal distance between 2 consecutive points of the snake and N the total number of points of each part of the snake. In this paper, these parameters have been tuned to the following values: $(\theta_{\inf}, \theta_{\sup}, N, \Delta) = (-\frac{\pi}{3}, \frac{\pi}{5}, 6, 5)$. For the definition of the angular sector, the condition $|\theta_{\inf}| > |\theta_{\sup}|$ has to be verified in order to make the jumping snake go towards the mouth. The values of

N and Δ are the outcome of a compromise between accuracy of the contour and convergence speed.

4.3.3 Key points detection.

The key points give important cues about the lip shape. They are used as fulcra for the computation of the model. We use six principal key points (see Figure 2): the right and left mouth corners (Q_1 and Q_5), the lower central point (Q_6) and the three points of the Cupidon's bow (Q_2, Q_3 and Q_4). We also use two secondary points located inside the mouth: Q_7 and Q_8 (see Figure 16). They are used to find the lower central point Q_6. Moreover, they will be useful in a future work for the inner lip boundary segmentation.

The three upper points are located on the estimated upper lip boundary resulting from the jumping snake algorithm. Q_2 and Q_4 are the highest points on the left and right of the seed. Q_3 is the lowest point of the boundary between Q_2 and Q_4 (see Figure 16).

The points Q_6, Q_7 and Q_8 are found by analyzing $\nabla_y(h)$, the 1D gradient of the pseudo hue along the vertical axis passing by Q_3 (see Figure 16). The pseudo hue is higher for the lips than for skin or teeth and tongue. So, the maximum of $\nabla_y(h)$ below the upper boundary gives the position of Q_7. Q_6 and Q_8 are the minima of $\nabla_y(h)$ below and above Q_7 respectively. This requires a face alignment with the vertical so that lips are horizontally aligned.

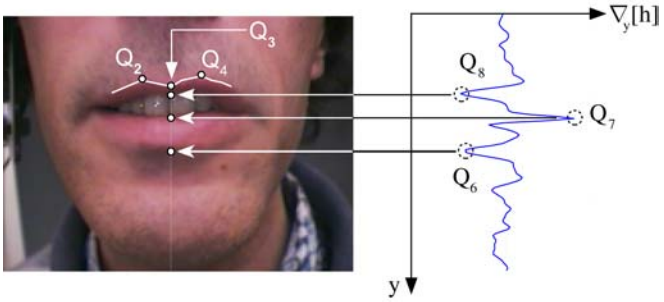


Figure 16: The 3 upper points are found on the estimated upper boundary resulting from the jumping snake algorithm (white line). Q_6, Q_7 and Q_8 are below Q_3, on extrema of $\nabla_y[h]$.

5. Models fitting

5.1 Eyes and eyebrows

Figure 17 gives an example of initial models for eye and brow.

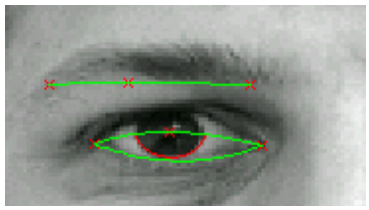


Figure 17: Initialization of eye and brow contours.

The process for model fitting remains the same: each contour is supposed to be made of a set of points with maximum of luminance gradient. The curve to be selected maximizes the *NFLG* through the contour.

Points (P_1, P_2 and P_4) are detected with enough accuracy so that the lower eyelid parabola has not to be modified again. On the contrary, since the P_3 control point of the upper Bezier curve is initialized at the iris centre, we know that this initial curve has to be adjusted. Point P_3 moves upwards along a vertical axis (points P_1 and P_2 being fixed) until the *NFLG* through the contour is maximum.

For the fitting of the brow model, the same idea is used since brow contour is made of points with local maximum of luminance gradient. The main difference concerns the strategy of points displacement. The three control points P_5, P_6 and P_7 are moved (along a vertical axis only for P_5 and P_7, and along all the possible upward directions for P_6).

5.2 Mouth

The deformation strategy of the cubic curves based mouth model is different because model fitting and mouth corners detection are tightly linked. If a human operator has to find the corners, he implicitly uses the global shape of the mouth. He follows the upper and lower edges, extends them when they are becoming indistinct, and finally put the corner where they intersect. So corners and boundaries are found in a single operation. We propose an algorithm that works the same way.

Basically, a cubic curve is uniquely defined if its four parameters are known. Here, each curve passes by, and has a null derivative on points Q_2, Q_4 or Q_6. These considerations bring 2 constraint equations that decrease the number of parameters to be estimated from 4 to 2 for each cubic. So, only two more points of each curve are needed to achieve the fitting. These missing points are chosen in the most reliable parts of the boundary, i.e. near Q_2, Q_4 or Q_6. Upper curves missing points have already been found by the *jumping snake*. On the other hand, only one point (Q_6) of the lower boundary is known. To get additional lower points, we make a snake grow from the seed Q_6. The growth stops after a few points (the white dots in Figure 18).

Now that there are enough points for each part of the boundary, it should be possible to compute the

curves γ_i passing by them and to find the mouth corners where these curves intersect.

In order to make results more robust, we also suppose that the corners (Q_1, Q_5) are known. This consideration brings another constraint equation that decreases the number of parameters to be estimated from 2 to 1 for each curve. Therefore, the least squares minimization is achieved very quickly.

Moreover, the resulting curves are much less sensitive to supplementary chosen points positions. In other words, a given corner position corresponds to a unique and easily computed set of curves. So, the fitting is achieved by finding the corners that give the best curves. Obviously, an exhaustive test over all the pixels of the image would be too long. But, a very simple hypothesis can help reducing the search area to several pixels. We consider that the mouth corners are located in dark areas. So, we look for the minima of luminance for each column between the upper and the lower boundary. It gives the line of minima L_{mini} (see Figure 18). We suppose that the corners are located on this line. A given (right or left) corner corresponds to a unique couple of upper and lower curves (the dashed curves in Figure 18). So, the fitting is achieved by finding the corners that gives the best couple of curves.

To know if a curve fits well to the lip boundary, we use an edge criterion. The model fitting consists in finding the corners, which are going to provide the closest cubic curves to the real contours. Real contours are characterized by a maximal R_{top} flow.

If the upper curves γ_1 and γ_2 fit perfectly to the edge, they are orthogonal to the $\vec{R}_{top}$ gradient field. On the other hand, the curves γ_1 and γ_2 have to be orthogonal to the $\vec{\nabla}[h_N]$ gradient field. We compute $\phi_{top,i}$ and $\phi_{low,i}$, the mean flows through the upper and lower curves respectively:

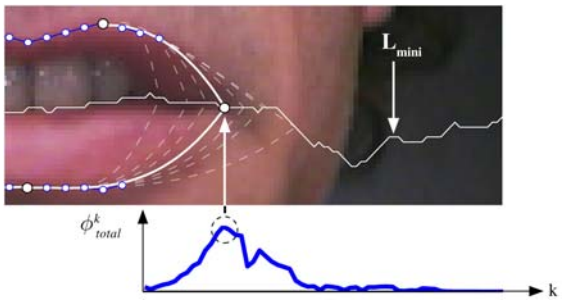


Figure 18 the maximum of ϕ^k_{total} gives the position of the corner along L_{mini}. The dotted curves are the cubic curves associated to the different *k* tested points along L_{mini}.

$$\phi_{top,i} = \frac{\int_{\gamma_i} \vec{R}_{top} . d\vec{n}}{\int_{\gamma_i} ds} \quad i \in \{1,2\}$$

$$\phi_{low,i} = \frac{\int_{\gamma_i} \vec{\nabla}[h_N] . d\vec{n}}{\int_{\gamma_i} ds} \quad i \in \{3,4\}$$

where $d\vec{n}$ and ds are the vector orthogonal to the segment and the curvilinear abscissa respectively. We consider n possible positions along L_{mini} for each point Q_1 and Q_5 respectively. The best position gives a high $\phi_{top,i}$ and a very negative $\phi_{low,i}$. So, on each side we have to maximize ϕ_{total}, computed as follows:

$$\phi^k_{total} = \phi^k_{top,normalized} - \phi^k_{low,normalized}, \quad k \in \{1,\ldots,n\}$$

with:

$$\phi^k_{top,normalized} = \frac{\phi^k_{top} - \min_{j\in\{1,\ldots,n\}} \{\phi^j_{top}\}}{\max_{j\in\{1,\ldots,n\}} \{\phi^j_{top}\} - \min_{j\in\{1,\ldots,n\}} \{\phi^j_{top}\}} \quad (7)$$

ϕ^k_{top} and ϕ^k_{low} are associated to the tested corner number *k*. $\phi^k_{top,normalized}$ and $\phi^k_{low,normalized}$ are their normalized value over the whole tested set. When ϕ^k_{total} is high, the corner position is reliable because the corresponding curves fit well to the lip boundaries. So the boundaries and the corners are found in a single operation. Figure 18 shows the evolution of ϕ^k_{total} for different values of *k*. The maximum of ϕ^k_{total} gives the position of the corner along L_{mini}.

6. Results

The segmentation method has been firstly tested on a video database acquired at our laboratory. Each video has been acquired with a digital camera and the filmed person was more or less far from the camera. There is no constraint about the lighting conditions (so far as possible as "natural" conditions). The FERET database [10] (with more than 3500 images) and the Kanade and Cohn database [4] have also been used for tests. But for the Kanade and Cohn images, only eyes and brows have been extracted since the data are only luminance data while mouth contour segmentation requires chrominance information.

The choice of a database for testing is a difficult task. In [14], the MIT and CMU databases are presented. In [27], 8 other face databases are described. Another difficulty is that all these

databases have been defined for face recognition and detection. Images are not always adapted to our problem of facial features extraction since some features can be hidden or some faces are too small in the frame in order to proceed to facial features extraction.

6.1 Models pertinence and quality segmentation

Figure 19 allows visually evaluating the pertinence and the flexibility of the chosen model for the outer mouth contour. The choice of the right mouth model is a great challenge because mouth is highly deformable. With the proposed model, it is possible to segment very different mouth shapes even in case of a grimace. This is very important with applications such as vocal message understanding improvement in case of noisy transmission. Indeed, the use of visual information in order to increase the intelligibility of a message is efficient only if this information is very accurate (especially, the extracted contours should be very precise). The accuracy of the proposed algorithm is to be related chiefly to the chosen models for mouth, eyes and brows. Indeed, the choice of a model allows putting some regularization constraints on the solution.

Figure 19: segmented mouth contours in case of a grimace.

Figure 20 shows examples of segmentation results for the whole considered facial features. Segmented contours are of good quality even if the subject is wearing spectacles. Spectacles are not a problem since the first extraction concerns the iris semi circle detection and spectacles are in general bigger than the eyes. As far as the brow segmentation is concerned, it is possible that in some cases, the brow contour coincides with the spectacles one. But in such cases, it is not possible to make the difference even visually.

In order to have a quantitative evaluation of the segmentation, a comparison with a manual ground truth has been done. The accuracy is estimated for the key points Q_1, …, Q_6 of the mouth, for the key points P_1, P_2 for each eye and for the key points P_5 , P_7 for each brow.

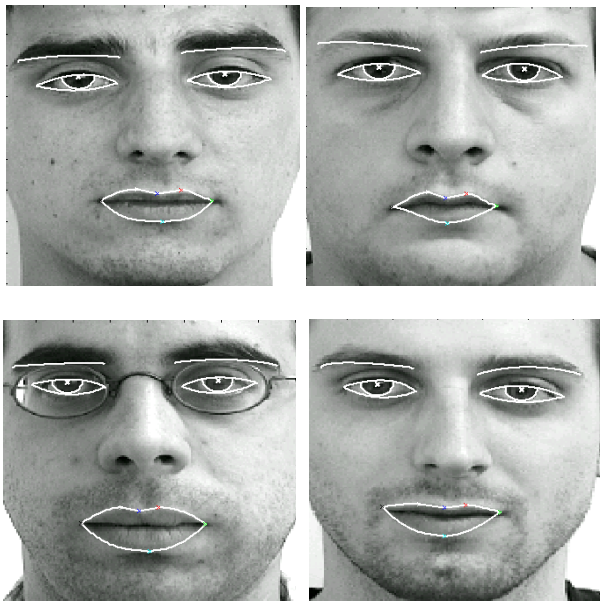


Figure 20: extracted contours for different persons.

For this comparison, 300 frames coming from 8 different sequences have been manually labelled by 5 different persons on the one hand and on the other hand, they have been segmented by using the proposed algorithm. Figure 21 gives a representation of the average error for each point: the radius of each circle is directly proportional to the error rate. Mouth corners are extracted with less accuracy (these points are sometimes very difficult to identify even visually). Concerning the eyes and the brows, the accuracy is lower for the external corners, which are more difficult to characterize than internal corners.

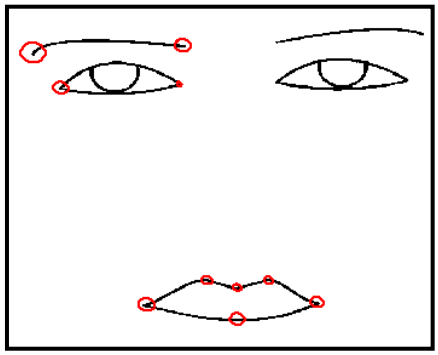


Figure 21: Quantitative evaluation of the segmentation accuracy.

The numerical values of accuracy are summarized in Table 1. Errors are relative to the mouth width, eyes width and brows width respectively. For a purpose of comparison, we also give the dispersion obtained during the manual labelling by different persons. Even with a manual labelling, the relative error is non null (cf. Table 1) and this error is higher for the external points of eyes and brows (these points are more difficult to detect even visually).

	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	P_1	P_2	P_5	P_7
P_{algo}	4.5	2.9	2.3	2.4	3.8	4.1	6.1	2.9	7.8	4.9
P_{hand}	1.5	1.1	1.1	1.2	1.6	1.6	2.5	1.7	3.2	1.9

Table 1: relative errors (in %) after the automatic extraction (P_{algo}) and the manual extraction (P_{hand}) of the key points.

For a more complete quantitative study of the quality of the mouth segmentation, two approaches could be considered: clone lip motion synthesis and speech recognition (both approaches are under study). The accuracy of the segmentation should be related to the considered applications. For example, a good segmentation for facial expression recognition will be the segmentation, which leads to the best classification rates.

6.2 Application to emotion recognition

The extracted contours are used in a system of recognition of the six basic universal emotions (joy, surprise, sadness, disgust, anger, fear). From a physiological point of view, facial expressions result from facial features deformations. This approach has been validated by a rate of 60% of good recognition obtained by an experimentation carried out in psychology where 60 judge subjects (30 males and 30 females) had to recognized an expression by only viewing the facial contours.

The classification is based on the study of the temporal evolution of the facial features skeletons. Figure 22 gives examples of emotional skeletons in case of joy and surprise. The considered distances for the fusion process are defined on the same figure.

A mouth opening and an eye wrinkling characterize Joy emotion. A full eye opening and a moving up motion of the brows characterize surprise emotion.

The description of the recognition system using the distances D_i in a fusion process based on the belief theory is given in [12].

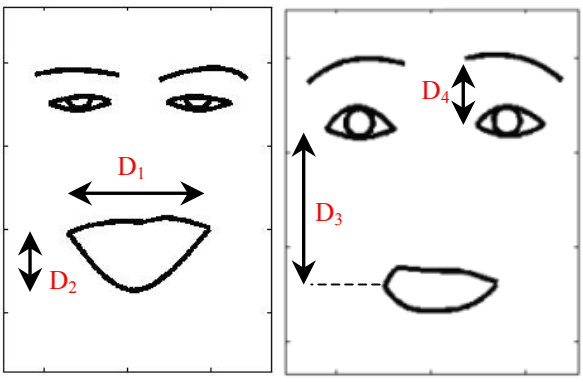


Figure 22: emotion skeletons for *joy* (on the left) and surprise (on the right); characteristic distances.

6.3 About algorithms parameters

The proposed algorithms involve the following parameters:

- For the mouth, 4 parameters $(\theta_{inf}, \theta_{sup}, N, \Delta)$ are used for the jumping snake definition. The values of these parameters are related to the mouth size and to a compromise between accuracy and complexity [9]. The used values for all the presented examples are given in section 4.3.2
- For eyes and brows, the parameter to be manually estimated in the first frame is the iris radius.

As far as the processing rate is concerned, the extraction of the contours of mouth, iris and eyes is done at a frame rate of 15 frames per second with a C code implementation on a standard low cost PC and for the processing of QCIF images. The brow segmentation is much slower because the three control points have to be adjusted during the fitting step. A preliminary study done at the laboratory about the implementation on a system on chip of our algorithm for the lips and eyes segmentation shows that real time processing is achievable [17].

We are working on the problem of head pose estimation in order to alleviate the constraint of vertical head. But the pertinence of the proposed parametric models have been shown: the models are enough flexible to describe the considered contours with accuracy.

7. Conclusion

In this article, parametric models for the purpose of permanent facial features segmentation have been proposed. The extracted contours are enough accurate to be used in an application dedicated to a high level interpretation such as facial expression recognition.

References

[1] P.S. Aleksic, J.J Williams, Z. Wu et K. Katsaggelos "Audio-Visual Speech Recognition Using MPEG-4 Compliant Visual Features" *EURASIP Journal on Applied Signal Processing, Special Issue on Joint Audio-Visual Speech Processing*, September 2002, pp.1213-1227.

[2] W. Beaudot. The neural information processing in the vertebra retina : a melting pot of ideas for artificial vision. *Phd thesis, tirf laboratory*, Grenoble, France, 1994.

[3] M.O. Berger, R. Mohr „Towards Autonomy in Active Contour Models“ *In Proc. ICPR'90*, June 1990, pp.847-851.

[4] Kanade and Cohn emotional video sequences URL : http://www.cs.cmu.edu/~face

[5] T. Coianiz, L Torresani , B. Caprile "2D Deformable Models for Visual Speech Analysis" *In NATO Advanced Study Institute : Speech reading by Man and Machine*, 1995, pp.391-398.

[6] T.F Cootes, C.J. Taylor, D. Cooper and J. Graham "Active Shape Models: their Training and Application" *In Computer Vision and Image Understanding*, 1(61), pp.38-59, January 1995.

[7] N. Eveno, A. Caplier, P.Y. Coulon "A new color transformation for lip segmentation" *In Proc. IEEE MSSP'01*, Cannes, France, September 2001.

[8] N. Eveno, A. Caplier, P.Y. Coulon. A parametric model for realistic lip segmentation. *International Conference on Control, Automation, Robotics and Vision (ICARV'02)*, Singapore, December 2002.

[9] N. Eveno, A. Caplier, P.Y. Coulon. "Jumping snakes and parametric model for lip segmentation". *International Conference on Image Processing*, Barcelone, Espagne, Septembre 2003.

[10] FERET database : http://www.itl.nist.gov/iad/humanid/feret/feret_master.html

[11] The ORL Database of Faces, Cambridge University Department, http://www.uk.research.att.com/facedatabase.html

[12] Z. Hammal, A. Caplier, M. Rombaut -Belief Theory Applied to Facial Expressions Classification- *3rd International Conference on Advances in Pattern Recognition*, Bath, United Kingdom, August 2005.

[13] M.E. Hennecke, K.V Prasad, D.G. Storck "Using Deformable Templates to Infer Visual Speech Dynamics". *In Proc. 28th Annual Asilomar Conference on Signals, Systems and computers*, 1994, pp. 578-582.

[14] H. Hjelmäs, B. Low "Face detection: a survey". *Computer Vision and Image Understanding*, 83, 2001, pp. 236-274.

[15] A. Hulbert, T. Poggio "Synthesizing a Color Algorithm from Examples" *Science*, Vol.239, 1998, pp.482-485.

[16] M. Kass, A. Witkins, D. Tersopoulos „Snakes : Actives Contours Models“, *International Journal of computer vision, 1(4),* January 1988, pp.321-331.

[17] S. Mancini and N. Eveno. "An IIR based 2D adaptive and predictive cache for image processing". In P. Fouillat, editor, *DCIS 2004*, Bordeaux, France, November 2004.

[18] A. Nefian, L. Liang, X. Pi, L. Xiaoxiang, C. Mao et K. Murphy « A couple HMM for Audio-visual Speech Recognition ». *In Proc. ICASSP'02*, 2002, pp.2013-2016.

[19] K. Sobottka and I. Pitas. "Looking for Faces and Facial Features in Color Images", *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications, Russian Academy of Sciences*, Vol. 7, No. 1, 1997.

[20] D. Terzopoulos et K. Waters « Analysis and Synthesis of Facial Image Sequences Using Physical and Anatomical Models » *IEEE Trans. On PAMI*, 15(6), June 1993, pp.569-579.

[21] Y. Tian, T. Kanade, J. Cohn "Robust Lip Tracking by Combining Shape, Color and Motion". *Proc ACCV'00*, 2000.

[22] Y. Tian, T. Kanade, and J.Cohn "Dual state Parametric Eye Tracking". *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition* , Grenoble, France, pp. 110 – 115, March 2000.

[23] N. Tsapatsoulis, Y. Avritis, S. Kollias « Efficient Face Detection for Multimedia Applications ». *In Proc. ICIP'00*, Vancouver, Canada, September 2000.

[24] P. Viola, J. Jones « Robust Real Time Face Detection » *International Journal of Computer Vision*, 57(2):137-154, May, 2004

[25] R. Wang and Y. Wang, "Facial Feature Extraction and Tracking in Video Sequences" *IEEE Signal Processing Society 1997 Workshop on Multimedia Signal Processing* June 23 --- 25, 1997, Princeton, New Jersey, USA Electronic Proceedings. pp. 233-238.

[26] M.H. Yang, N. Ahuja « Gaussian Mixture Model for Human Skin Color and Its Application in Image and Video Database ». *In Proc. Of the SPIE: Conf. On Storage and Retrieval for Imges and video Databases*, vol.3656, 1999, pp.458-466.

[27] M.H. Yang, D. Kriegman, and N. Ahuja "Detecting face in images : a survey". *IEEE Trans on PAMI*, vol.24, n°1, pp. 34-58, January, 2002.

[28] A. Yuille, P. Hallinan et D. Cohen « Feature Extraction from faces using deformable templates » *Int. Journal of computer Vision*, 8(2), 1992, pp.99-111.

[29] X. Zhang, R.M. Mersereau, M.A. Clements, C.C Broun « Visual Speech Feature Extraction for Improved Speech Recognition" *In Proc. ICASSP'02*, 2002, pp. 1993-1996.

[30] Machine Perception Toolbox, face detection algorithm: http://mplab.ucsd.edu/grants/project1/free-software/MPTWebSite/introductionframe.html