

A Novel Biometric System Based on Lips

K. Decika¹, M. Mohanraj²

¹Assistant Professor, Maharaja Co-Educational Arts and Science College, Erode- India

²Research Scholar, Maharaja Co-Educational Arts and Science College, Erode- India

E-mail- mohanraj.manoleela@gmail.com

Abstract:

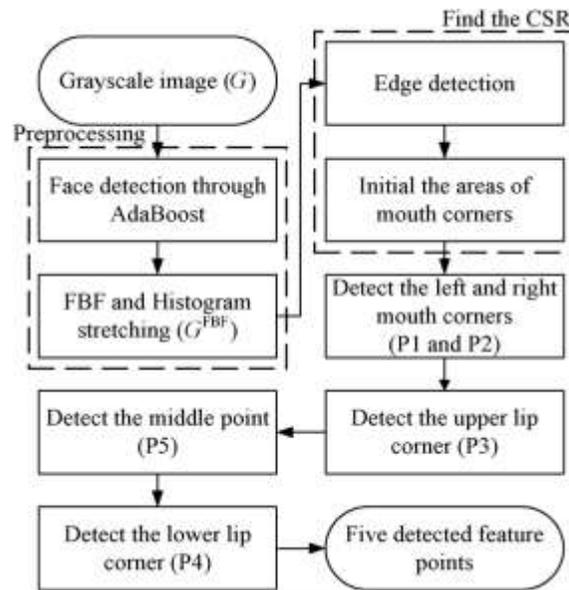
In this paper, the A Novel Biometric System based on Lips for identity recognition is investigated. In fact, it is a challenging issue for identity recognition solely by the lips. In the first stage of the proposed system, a fast box filtering is proposed to generate a noise-free source with high processing efficiency. Afterward, five various mouth corners are detected through the proposed system, in which it is also able to resist shadow, beard, and rotation problems. For the feature extraction, two geometric ratios and ten parabolic-related parameters are adopted for further recognition through the support vector machine.

Index Terms— Lip tracking, localized colour active contour, localized energy, deformable model.

1. INTROUCTION

Detecting lip contour with high accuracy is an important requirement for a lip identification system, and it has been widely discussed in former works. One of the studying directions considers the colour information. Hosseini and Ghofrani's work and the method presented in [2] converted a RGB colour image into $CIEL^*U^*V^*$ and $CIEL^*a^*b^*$ color spaces. Components a^* and U^* were combined together to generate a new image to emphasize the lip features, and the results were analyzed through the 2-D fast wavelet transform. Finally, the morphology was employed to smooth and binarize the image and then removed the noises to obtain the lip region. The images were converted into Caetano and Barone's chromatic colour space. Afterward, the mean and the threshold were computed from the red channel of each pixel, and these were employed to separate the lips' and nonlips' regions. A new colour mapping method, which integrated colour and intensity information, was developed for the lips' contour extraction, and Otsu's thresholding was adopted to extract the binary result.

Mouth Corner Detection:



2.Pre-processing:

In this paper, the face region is directly detected by the powerful method, namely, Viola and Jones' face detection algorithm from an image, an example of which is shown in figure with the red box. For further refining the possible region of the lips, a subregion is roughly extracted by the following estimation:

$$\begin{aligned}(i^2, j^2) &= (i^0, j^0 + 0.25 \times M) \\ (i^3, j^3) &= (i^1 - N \times 0.6, j^1 - M \times 0.25)\end{aligned}$$

Where (i^0, j^0) and (i^1, j^1) denote the origin and the top-right position of the face's bounding box of size $M \times N$; positions (i^2, j^2) and (i^3, j^3) denote the origin and the top-right position of the estimated lip region of size $M \times N$.

For easing the influences from the camera noise and various lighting changes, and achieving a lower computation complexity simultaneously, the proposed fast box filtering (FBF) and the well-known histogram stretching method are used to obtain a contrast enhanced and smoothed result. The first step of the proposed FBF method is to obtain an integral image as derived by

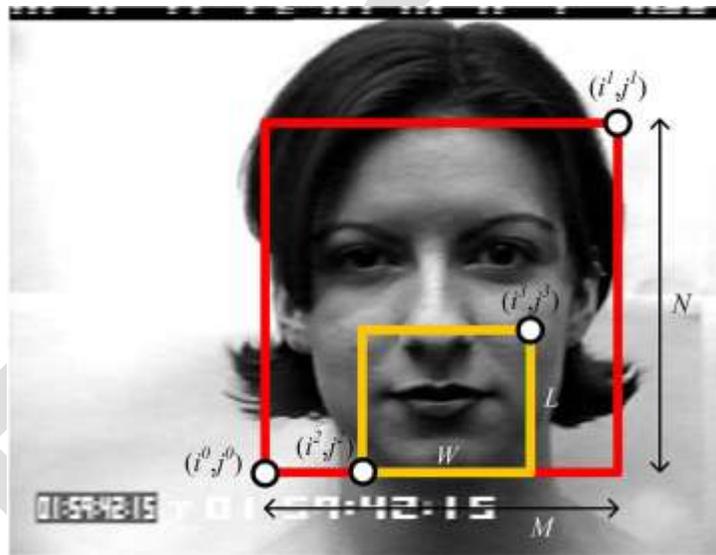
$$r_{i,j} = g_{i,j} + r_{i,j-1} + r_{i-1,j} - r_{i-1,j-1}$$

Where $r_{i,j}$ and $g_{i,j} \in G$ denote the integral value and the original gray-scale value, respectively. Each of the obtained $r_{i,j}$ denotes the summation result of the whole gray-scale values within its bottom-left region. Afterward, the response of the box filtering (BF) can be obtained by the following calculation:

$$b_{i,j} = \left[r_{i+\lceil \omega/2 \rceil, j+\lceil \omega/2 \rceil} - r_{i-\lceil \omega/2 \rceil, j+\lceil \omega/2 \rceil} - r_{i+\lceil \omega/2 \rceil, j-\lceil \omega/2 \rceil} + r_{i-\lceil \omega/2 \rceil, j-\lceil \omega/2 \rceil} \right] / \omega^2$$

Where $b_{i,j}$ denotes the smoothed result, the odd parameter ω denotes the size of the employed filter size, and notations $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote the round-up and round-down operators, respectively.

The rough lip region (bounding with the obtained (i^2, j^2) and (i^3, j^3) positions) is processed with the proposed FBF and the histogram stretching method. The corresponding region, as shown in figure is adopted as an example to exhibit the result, and the enhanced smoothed image with $\omega = 3$ is shown in figure. This image is named G^{FBF} in this paper, which is widely used in this paper.



3. Understanding Support Vector Machines

- Separable Data
- Non-separable Data
- Nonlinear Transformation with Kernels

Separable Data

You can use a support vector machine (SVM) when your data has exactly two classes. An SVM classifies data by finding the best hyperplane that separates all data points of one class from those of the other class. The best hyperplane for an SVM means the one with the largest margin between the two classes. Margin means the maximal width of the slab parallel to the hyperplane that has no interior data points.

The support vectors are the data points that are closest to the separating hyperplane; these points are on the boundary of the slab. The following figure illustrates these definitions, with + indicating data points of type 1, and - indicating data points of type -

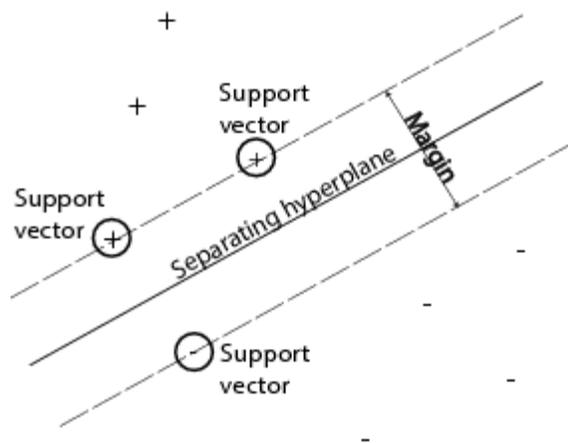


Fig (1)

Mathematical Formulation: Primal. This discussion follows Hastie, Tibshirani, and Friedman [12] and Christianini and Shawe-Taylor.

The data for training is a set of points (vectors) x_i along with their categories y_i . For some dimension d , the $x_i \in \mathbb{R}^d$, and the $y_i = \pm 1$. The equation of a hyperplane is

$$\langle w, x \rangle + b = 0,$$

Where $w \in \mathbb{R}^d$, $\langle w, x \rangle$ is the inner (dot) product of w and x , and b is real.

The following problem defines the best separating hyperplane. Find w and b that minimize $\|w\|$ such that for all data points (x_i, y_i) ,

$$y_i(\langle w, x_i \rangle + b) \geq 1.$$

The support vectors are the x_i on the boundary, those for which $y_i(\langle w, x_i \rangle + b) = 1$.

For mathematical convenience, the problem is usually given as the equivalent problem of minimizing $\langle w, w \rangle / 2$. This is a quadratic programming problem. The optimal solution w, b enables classification of a vector z as follows:

$$\text{class}(z) = \text{sign}(\langle w, z \rangle + b).$$

Mathematical Formulation: Dual. It is computationally simpler to solve the dual quadratic programming problem. To obtain the dual, take positive Lagrange multipliers α_i multiplied by each constraint, and subtract from the objective function:

$$L_P = \frac{1}{2} \langle w, w \rangle - \sum_i \alpha_i (y_i (\langle w, x_i \rangle + b) - 1),$$

where you look for a stationary point of LP over w and b . Setting the gradient of LP to 0, you get

$$w = \sum_i \alpha_i y_i x_i$$
$$0 = \sum_i \alpha_i y_i.$$

Substituting into LP, you get the dual LD:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle,$$

Which you maximize over $\alpha_i \geq 0$. In general, many α_i are 0 at the maximum. The nonzero α_i in the solution to the dual problem define the hyperplane, as seen in Equation 16-1, which gives w as the sum of $\alpha_i y_i x_i$. The data point's x_i corresponding to nonzero α_i are the support vectors.

The derivative of LD with respect to a nonzero α_i is 0 at an optimum. This gives

$$y_i (\langle w, x_i \rangle + b) - 1 = 0.$$

In particular, this gives the value of b at the solution, by taking any i with nonzero α_i .

The dual is a standard quadratic programming problem. For example, the Optimization Toolbox™ quadprog solver solves this type of problem.

Nonseparable Data

Your data might not allow for a separating hyperplane. In that case, SVM can use a soft margin, meaning a hyperplane that separates many, but not all data points.

There are two standard formulations of soft margins. Both involve adding slack variables s_i and a penalty parameter C .

- The L1-norm problem is:

$$\min_{w,b,s} \left(\frac{1}{2} \langle w, w \rangle + C \sum_i s_i \right)$$

such that

$$y_i (\langle w, x_i \rangle + b) \geq 1 - s_i$$

$$s_i \geq 0.$$

The L1-norm refers to using s_i as slack variables instead of their squares. The SMO svmtrain method minimizes the L1-norm problem.

- The L2-norm problem is:

$$\min_{w,b,s} \left(\frac{1}{2} \langle w, w \rangle + C \sum_i s_i^2 \right)$$

subject to the same constraints. The QP svmtrain method minimizes the L2-norm problem.

In these formulations, you can see that increasing C places more weight on the slack variables s_i , meaning the optimization attempts to make a stricter separation between classes. Equivalently, reducing C towards 0 makes misclassification less important.

Mathematical Formulation: Dual. For easier calculations, consider the L1 dual problem to this soft-margin formulation. Using Lagrange multipliers μ_i , the function to minimize for the L1-norm problem is:

$$L_P = \frac{1}{2} \langle w, w \rangle + C \sum_i s_i - \sum_i \alpha_i (y_i (\langle w, x_i \rangle + b) - (1 - s_i)) - \sum_i \mu_i s_i,$$

where you look for a stationary point of LP over w , b , and positive s_i . Setting the gradient of LP to 0, you get

$$b = \sum_i \alpha_i y_i x_i$$

$$\sum_i \alpha_i y_i = 0$$

$$\alpha_i = C - \mu_i$$

$$\alpha_i, \mu_i, s_i \geq 0.$$

These equations lead directly to the dual formulation:

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$$

subject to the constraints

$$\sum_i y_i \alpha_i = 0 \\ 0 \leq \alpha_i \leq C.$$

The final set of inequalities, $0 \leq \alpha_i \leq C$, shows why C is sometimes called a box constraint. C keeps the allowable values of the Lagrange multipliers α_i in a "box", a bounded region.

The gradient equation for b gives the solution b in terms of the set of nonzero α_i , which corresponds to the support vectors.

You can write and solve the dual of the L2-norm problem in an analogous manner. For details, see Christianini and Shawe-Taylor [7], Chapter 6.

svmtrain Implementation. Both dual soft-margin problems are quadratic programming problems. Internally, svmtrain has several different algorithms for solving the problems. The default Sequential Minimal Optimization (SMO) algorithm minimizes the one-norm problem. SMO is a relatively fast algorithm. If you have an Optimization Toolbox license, you can choose to use quadprog as the algorithm. quadprog minimizes the L2-norm problem. quadprog uses a good deal of memory, but solves quadratic programs to a high degree of precision (see Bottou and Lin [2]). For details, see the svmtrain function reference page.

Nonlinear Transformation with Kernels

Some binary classification problems do not have a simple hyperplane as a useful separating criterion. For those problems, there is a variant of the mathematical approach that retains nearly all the simplicity of an SVM separating hyper plane.

This approach uses these results from the theory of reproducing kernels:

- There is a class of functions $K(x,y)$ with the following property. There is a linear space S and a function ϕ mapping x to S such that

$$K(x,y) = \langle \phi(x), \phi(y) \rangle.$$

The dot product takes place in the space S .

- This class of functions includes:

- Polynomials: For some positive integer d ,

$$K(x,y) = (1 + \langle x,y \rangle)^d.$$

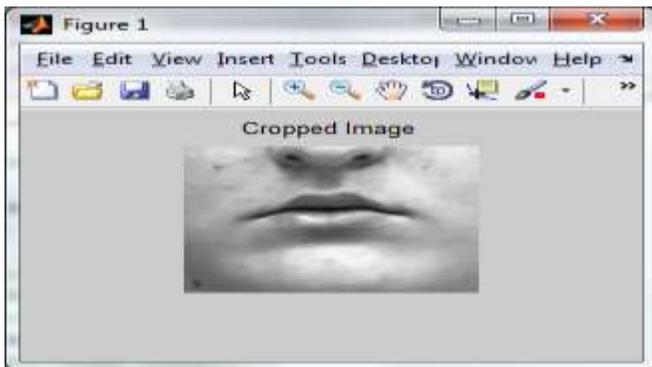
- Radial basis function: For some positive number σ ,

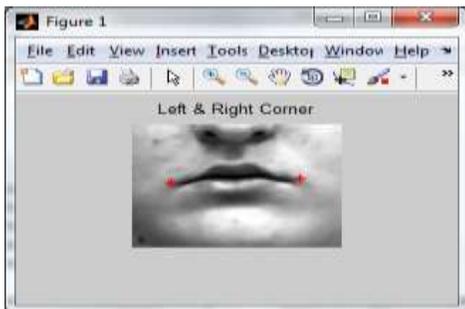
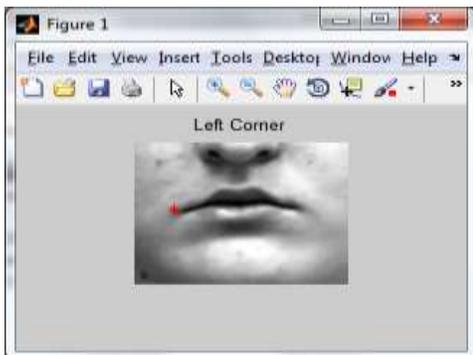
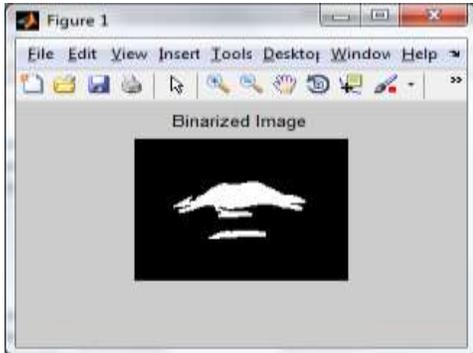
$$K(x,y) = \exp(-\langle x-y, x-y \rangle / (2\sigma^2)).$$

- Multilayer perceptron (neural network): For a positive number p_1 and a negative number p_2 ,

$$K(x,y) = \tanh(p_1 \langle x,y \rangle + p_2).$$

The mathematical approach using kernels relies on the computational method of hyperplanes. All the calculations for hyperplane classification use nothing more than dot products. Therefore, nonlinear kernels can use identical calculations and solution algorithms, and obtain classifiers that are nonlinear. The resulting classifiers are hyper surfaces in some space S , but the space S does not have to be identified or examined.





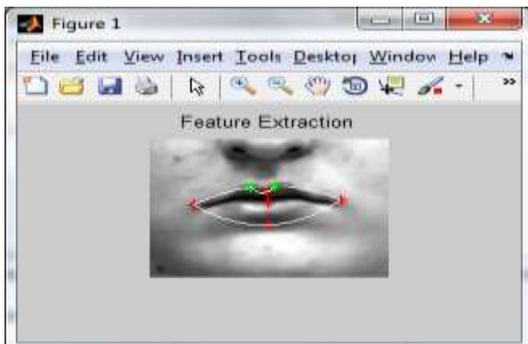
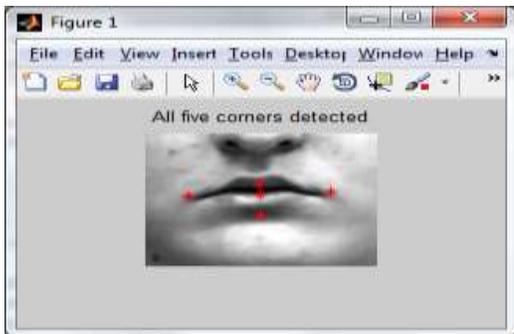
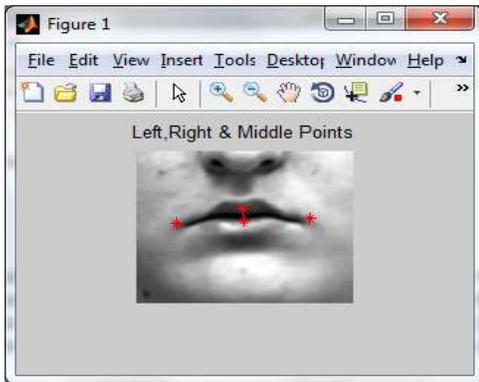
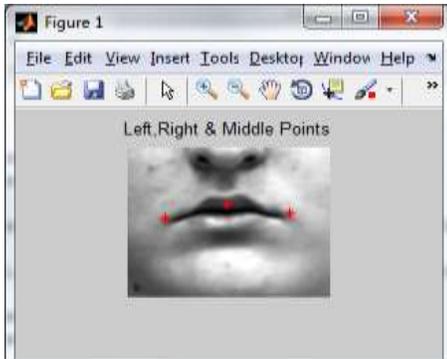


Fig.1 . Tracking result using the proposed method.

Table 1. Computing time of the proposed algorithm.

Algorithm Step	Extracting	Tracking
Iteration [average]	25	5
Computing time[average]	0.695s	0.103s

Tracking results are shown in Fig. 1 and Table 1.

It is found that the proposed algorithm has achieved a promising tracking result, which is robust against the appearance of teeth and tongue. Furthermore, the utilization of a 16-point deformable model to describe a lip shape is physically meaningful. In addition, the computing time of tracking one lipframe is less than the extracting process. In particular, when there exists a long lip sequences, it is effective to utilize the previous lip contour as the important parameter to track the current one. It is expected that such an operation can reduce a large amount of computing time.

5. CONCLUSION

In this paper, we have proposed a robust lip tracking algorithm using localized colour active contours and deformable models. This approach is adaptive to the lip movements, and also robust against the appearance of teeth and tongue. Hence, it provides a promising way for lip tracking.

REFERENCES:

- [1] M.U. Ramos Sánchez, J. Matas, and J. Kittler, "Statisticalchromaticity-based lip tracking with b-splines," in Proceedings of ICASSP'1997, 1997, vol. 4, pp. 2973–2976.
- [2] A.W.C. Liew, S.H. Leung, and W.H. Lau, "Lip contourextraction from color images using a deformablemodel," Pattern Recognition, vol. 35, no. 12, pp. 2949–2962, 2002.
- [3] G.I. Chiou and J.N Hwang, "Lipreading from colorvideo," IEEE Transactions on Image Processing, vol. 6,no. 8, pp. 1192–1195, 1997.
- [4] I. Matthews, T. F. Cootes, J. A. Bangham, S. Cox, andR. Harvey, "Extraction of visual features for lipreading,"IEEE Transactions on Pattern Analysis and Machine Intelligence,vol. 24, no. 2, pp. 198–213, 2002.[5] N. Eveno, A. Caplier, and P.Y. Coulon, "Accurate andquasi-automatic lip tracking," IEEE Transactions on Circuitsand Systems for Video Technology, vol. 14, no. 5,pp. 706–715, 2004.
- [6] S.L. Wang, W.H. Lau, and S.H. Leung, "Automatic lipcontour extraction from color images," Pattern Recognition,vol. 37, no. 12, pp. 2375–2387, 2004.[7] S. Lankton and A. Tannenbaum, "Localizing regionbasedactive contours," IEEE Transactions on Image Processing,vol. 17, no. 11, pp. 2029–2039, 2008.[8] M. Li and Y.M. Cheung, "Automatic lip localization underface illumination with shadow consideration," SignalProcessing, vol. 89, no. 12, pp. 2425–2434, 2009