Pseudoautomatic Lip Contour Detection Based on Edge Direction Patterns

Mihaela Gordan* Constantine Kotropoulos Ioannis Pitas
Department of Informatics, Aristotle University of Thessaloniki
Artificial Intelligence and Information Analysis Laboratory
GR-54006 Thessaloniki Box 451, Greece
{mihag,costas,pitas}@zeus.csd.auth.gr

Abstract

Detection and tracking of the lip contour is an important issue in speechreading. While there are solutions for lip tracking once a good contour initialization in the first frame is available, the problem of finding such a good initialization is not yet solved automatically, but done manually. Solutions based on edge detection and tracking have failed when applied on real world mouth images. In this paper we propose a solution to the lip contour detection that minimize the user interaction. A minimal number of points to be manually marked on the mouth image is required. The method is based on the examination of gradient direction patterns in the lips area, and makes use of the local direction constancy along the lip contours, as opposed to the other regions of the mouth image that are characterized by random edge directions.

1. Introduction

A relatively large class of lipreading algorithms are based on lip contour analysis. Examples of such algorithms can be found in [1, 2, 3]. In these cases, lip contour extraction is needed as the first step. By lip contour extraction, we usually refer to the process of lip contour detection in the first frame of an audio-visual image sequence. Obtaining the lip contour in subsequent frames is usually referred as lip tracking. While for lip contour tracking there are well-developed techniques and algorithms to perform this task automatically, in the case of lip contour extraction in the first frame the things are different. This is a much more difficult task than tracking, due to the lack of a good a-priori information in respect to the mouth position in the image, the mouth size, the approximate shape of the mouth, mouth opening etc. So, while in lip contour tracking we have a good initial estimate of the mouth contour from the previous frame, this initial estimate is not always available for the first frame, but it has to be produced by some means.

Different authors tried different procedures to solve the extraction of a good lip contour in the initial frame. Of course, the goal would be to solve this task automatically; approaches like region-based image segmentation and edge detection have been proposed. These methods work quite well in profile images and also in frontal images where the speaker wears lipstick or reflective markers. However, in the frontal images without any marking of the lips, the above-mentioned techniques unfortunately fail; and these images are the most used for speechreading. The problem of automatic extraction of the lip contour becomes even harder in the gray-level images, where the chromatic information differentiating between lips and skin is no longer present. Usually these images have a low contrast, so region-based segmentation and edge detection algorithms fail to provide good results [1], [4]. In these cases, the solution adopted is based on marking manually more or less points on the lip contour (or even on drawing manually the entire lip contour). When a large number of points (aprox. 50-100) are marked on the lip contour [4, 5], they are either used "as they are" to represent the lip contour (for example in lipreading based on Active Shape Models [6] and Active Appearance Models [3]), or an interpolation procedure is applied to obtain the entire lip contour (for example B-splines [5]). When a small number of points (e.g. 6 points) are marked on the lip contour, these points are used to derive some model parameters for the lip contour (for example the widely used ellipsoidal model [7] or parabolic model [8]). In the latter case, the accuracy of lip contour extraction is limited by the fitness of the model to the real lip contour. For example, in the case of an asymmetric mouth image (due let's say to a displacement of the video camera), the model-based lip contour representation might be different from the real lip contour. Such an example of mouth image is the one depicted in Figure 1 from the Tulips1 database [11].

*On leave from the Technical University of Cluj-Napoca, Faculty of Electronics and Telecommunications, Basis of Electronics Department, Cluj-Napoca, Romania

In this paper we propose a new approach to the problem of lip contour extraction in gray level images. The proposed approach is based on edge detection using gradient masks and edge following. The difference between the proposed solution and the previous edge-based methods, that makes our solution to work where the others failed, is the following: knowing that the mouth images have, typically, low contrast in the lip-to-skin area, and that false edges can appear in this area, we don't consider the edge magnitude as reliable information for lip contour detection. On the contrary, it is well known that the edge direction on the lip outlines will always be approximately piecewise constant and will follow a given pattern for all mouth images, while the "false edges" inside the lip and skin areas will have random, non-patterned directions. Taking this observation into account, we develop a piecewise edge following algorithm, having the constancy of the edge direction as main following criterion in each region. The experimental results obtained prove the good functionality of the proposed algorithm for outer lip contour extraction.

So far, the developed algorithm is semi-automatic, meaning that it requires to manually mark the start and end point of each sub-region of the mouth area, where the directional edge following is applied. Best results are obtained when the points are marked directly on a color map used for representing the edge directions, as shown in Section 2.

Although not completely automatic, the proposed algorithm has the advantage of providing a reliable lip contour without any geometric model assumption, while requiring a small number of points (6 to 12) to be manually labeled.

In a future work, the algorithm will be made completely automatic, by learning the directional patterns from a set of training mouth images and using the learned patterns to initialize edge following.

Figure 1. Example of an asymmetric mouth image from the Tulips1 database. [11]

2. Visual Representation of Edge Image Directions by Color Maps

As briefly explained in the previous section, the algorithm proposed here uses the edge property of each pixel from the mouth image to achieve the outer lip contour extraction. Since the quality of the lip contour extracted depends mainly on the edge follower applied in a later step, the edge detector can be very simple. In this case, we considered the use of the two Sobel convolution masks (horizontal and vertical):

$$h_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad h_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{1}$$

Two edge properties are used for representing every pixel: the edge (gradient) magnitude, denoted by $|\mathrm{g}|$, and the edge (gradient) direction, denoted by $\tan \alpha_g$. These two measures are computed from the horizontal and vertical gradient, denoted by g_x and respective g_y as in (2), resulting after convolving the 3×3 neighborhood of the current pixel with the horizontal and vertical Sobel masks.

$$\begin{aligned} |g| &= \sqrt{g_x^2 + g_y^2} \\ \tan \alpha_g &= g_y / g_x \end{aligned} \tag{2}$$

While in the case of representing the edges only by their magnitude (as is usually done in image processing applications) a gray level image representation is sufficient, in case we want to represent visually also the edge direction, we need one more image component, not only luminance. The most straightforward solution is to map the edge property of every pixel to a 3-component color, in which case we will have a color image for representing the edge image.

Since the edge direction is an angular measure, for a good edge magnitude-direction mapping in the color domain, we choose the HLS color space, where the hue H is represented as an angular measure. The luminance L will be used as an edge magnitude measure, thus maintaining the compatibility to the edge magnitude representation in classical edge detection schemes. The third component of the chosen color space, the saturation S, is not needed for mapping, so it will be set a-priori to its maximum possible value (i.e., we consider pure (saturated) colors). The formulas for edge to color mapping are:

$$\begin{aligned} L &= \frac{255 - |g|}{255}; \quad L \in [0,1]; \\ H &= 180 + \frac{\alpha_g}{\pi} \cdot 180; \quad H \in [0;360]; \\ \text{where } \alpha_g &= \arctan(\tan \alpha_g), \quad \alpha_g \in [-\pi, \pi] \\ S &= \begin{cases} 0 & \text{if } L = 1 \\ 1 & \text{if } L \neq 1 \end{cases} \end{aligned} \tag{3}$$

Due to the inherent low contrast of the mouth image, the edge magnitude as computed from the Sobel gradients g_x and g_y is very low. Therefore, to obtain a clear color representation of the edge image, an enhancement of the edge magnitude data is needed as a preprocessing step.

The block diagram of the edge detection and color mapping process is shown in Figure 2. The original mouth image, its edge magnitude image and the resulting color map

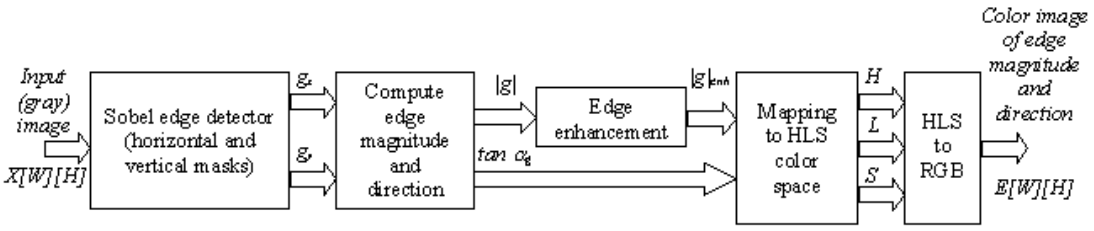

Figure 2. The processing chain for visual representation of edge magnitude and direction by color map

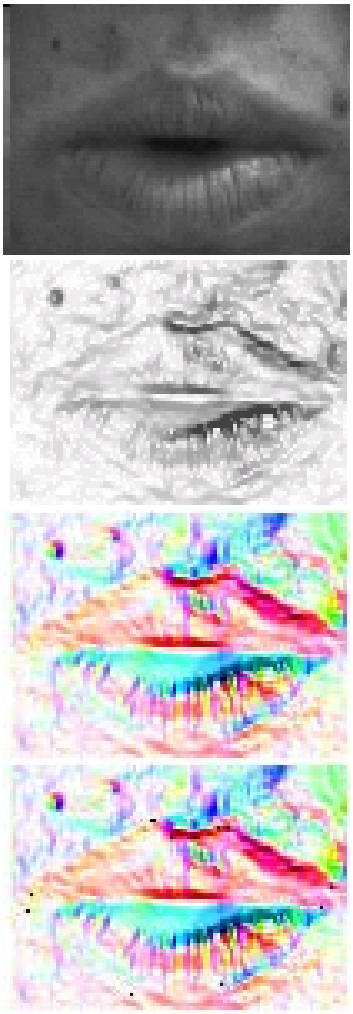

Figure 3. From top to botom: the original mouth image; the corresponding edge magnitude image; the resulting color map for edge magnitude and direction; the manually marked color map for defining the 6 non-overlapping lip contour regions

for the edge magnitude and direction in the case of a frame from Tulips1 database are shown in Figure 3.

Examining Figure 3, it is rather easy to verify the statement made in the previous paragraph: although the image of edge magnitudes (the edge image) is too poor to allow the detection of the lip contour, on the color map where the edge direction is also represented by hue one can easily observe the color patterns corresponding to the lips: there are 6 regions differentiable by their hue that correspond to the outer lip contour and are characterized by approximately constant hue inside each region. One can also notice that the manual marking of the start point and end point of each of these regions based on the color attribute is easy to be performed. The last image from Figure 3 shows an example of such a manual marking.

3. Description of the Lip Contour Extraction Algorithm

The lip contour extraction algorithm proposed here is based mainly on edge direction following.

For the time being, the algorithm is developed specifically for the outer lip contour extraction; however, the extraction of the inner lip contour can be performed in a similar way, by checking edge direction patterns in the inner area of the mouth.

The two main steps of the lip contour extraction algorithm are going to be described in the subsequent subsections.

3.1. Lip Sub-Region Identification

For performing a good outer lip contour extraction, the proposed algorithm starts with the computation of the edge magnitude and direction color map for the mouth image to be processed, as described in Section 2. On this color map, we require the manual plotting of 6 pairs of points, representing the start and end points of 6 rectangular regions identified on the mouth image. The lip contour extraction

will be performed piece-wise on each region; in the (small) remaining non-tracked areas we perform a linear contour interpolation for closing the lip contour. An example of such a manual marking of a mouth edge color map is shown in the last image of Figure 3.

We must note here that our algorithm doesn't eliminate completely the need of manual labeling (user intervention), but reduces the number of points that must be manually marked on the mouth image for obtaining a good contour quality as compared to the other non-model based lip contour extraction algorithms as it will be shown in Section 4.

Although the plot of the markers on the lips will not be done on "real" mouth images, but on the (artificially created) color maps, this doesn't represent a problem for the human operator, because the lip contour color patterns are clearly distinguishable on the color maps, as can be seen on the example shown in Figure 3.

3.2. The Directional Edge Following Algorithm

After the six distinct sub-image areas of the outer lip contour (characterized by a slightly varying edge direction) have been manually "specified" by the user, we apply an edge following algorithm based in principle on the heuristic search technique described in [9].

Most edge following algorithms take into account the edge magnitude as primary information for following. However, in the specific application of lip contour following for gray level mouth images, the edge magnitude information is not suitable, because it can be very weak in some contour areas and strong outside the mouth area or inside the lip region. This is exactly the reason why the algorithms that try to extract lip contour using classical edge detectors output fail [4]. In the specific case of lip contour extraction based on the edge property of the image pixels, the best information for following is given by the edge direction, as can be noticed from Figure 3. As a consequence, this information should be dominant in the cost function of the heuristic edge following algorithm in the case of lip contour detection.

The edge following algorithm used here is a modified version of the heuristic search algorithm described in [9]. There are two differences between the proposed edge following algorithm and the one from [9]:

a) The first difference regards the *cost function*. The general cost function of the heuristic edge following algorithm for a path connecting the start point x_1 to the end point x_N is a weighted sum of the individual costs of all the pixels in the path, and given by:

$$C(x_1, x_2, ..., x_N) = -\sum_{k=1}^{N} |g(x_k)| + a\sum_{k=2}^{N} |\alpha_g(x_k) - \alpha_g(x_{k-1})| + b\sum_{k=2}^{N} |g(x_k) - g(x_{k-1})|. \quad (4)$$

In our case, the cost function is based only on the edge direction, so only the second term from (4) is present in the cost function used, with the weighting coefficient a=1. But instead of using the absolute difference between the last pixel found in the path and the next candidate pixel, we use the absolute difference between *a mean direction of all the pixels found so far in the path* and the next candidate pixel. This way, some mean direction of the path is taken into account, and the probability for the following algorithm to take a (spurious) wrong direction is reduced. The cost function used and the computational formula for the mean direction are given in (5).

$$C(x_1, x_2, ..., x_N) = \sum_{k=1}^{N} |\alpha_{gmed}(k) - \alpha_g(x_k)|,$$

where

$$\alpha_{gmed}(k) = \begin{cases} \frac{\alpha_g(x_1) + \sum_{i=2}^{k-1} 2^{i-2} \cdot \alpha_g(x_i)}{2^{k-2}}, & k \geq 3, \\ \alpha_g(1), & k = 1, 2. \end{cases} \quad (5)$$

b) The second difference regards the search neighborhood implied: instead of using a 3 × 3 neighborhood for each of the 6 image sub-regions of the lip contour considered, we make use of the a-priori knowledge regarding the allowable directions for the lip contour piece-wise. We know that the lip contour can go only forward from the starting point to the ending point of each sub-region, so there is no point on searching the entire 3 × 3 neighborhood; only a 2 × 2 neighborhood will be enough, i.e., only 3 neighbor pixels of the current contour pixel are real possible candidates as next contour points. The search neighborhoods for each of the 6 lip contour sub-regions considered are given in Figure 4.

The overall algorithm for lip contour extraction based on the edge following can be described by the following steps:

START

• For each region r, $r = 1, \ldots, 6$,
Do:

Step 1. "Read" the starting point (x_{r_1}, y_{r_1}) and the ending point (x_{r_N}, y_{r_N})

Step 2. Select the directional 2 × 2 neighborhood of the region, denoted M_r to be used in the edge following algorithm.

Step 3. Apply the heuristic edge following:
for(r_k=r_1; r_k < r_N;r_k++)
Select:
x_{r_k}=arg min$_{j \in M_r(x_{r_{k-1}})}$ $C(x_{r_1},x_{r_2},...,x_{r_{k-1}},x_{r_j})$
where $C(x_{r_1},x_{r_2},...,x_{r_{k-1}},x_{r_j})$ is given by (5).

• Obtain the closed lip contour by linear interpolation between:
$((x_{r_N},y_{r_N}))$ and $(x_{r+1_1},y_{r+1_1}))$, for r=1, . . . , 5
and
$((x_{6_N},y_{6_N}))$ and $(x_{1_1},y_{1_1}))$

END

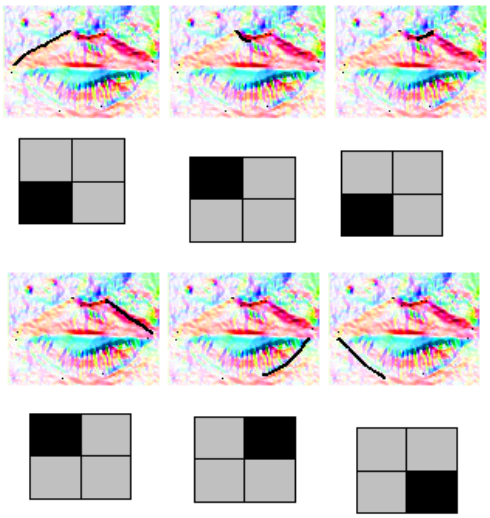

Figure 4. The six 2 × 2 search neighborhoods: from left to right: NE neighborhood; SE neighborhood; NE neighborhood; SE neighborhood; SW neighborhood; NW neighborhood

4. Experimental Results

For testing the performance of the proposed lip contour extraction algorithm, we used mouth images from the two most used databases in speechreading experiments: Tulips1 [11] and M2VTS [12]. The evaluation of the quality of the results was done visually.

In the case of tests on Tulips1 database we used:

• three different mouth images (frames) for the same subject (Anthony);

• other six different subjects (Ben; Candace; Cynthea; Regina; George; Oliver) with one mouth image per subject. The selection criterion was to have as big variation as possible between the mouth image properties in respect to: degree of mouth opening; symmetry/asymmetry of mouth image (e.g. Candace); mouth shape; lip-to-skin contrast. In all the cases, the lip contour extracted can be qualified as good (in some cases-very good).

In the case of M2VTS database, a rectangular region of interest containing the mouth was first selected manually from the face image. Then the lip contour extraction algorithm was applied to the selected mouth image. The results were compared to the similar ones reported by the Centre for Vision, Speech, and Signal Processing of the University of Surrey, using B-splines for lip contour detection [10]. Although there are some differences in the shape obtained, the extracted lip contour by our algorithm can be classified as good.

Some example images are given in Figure 5 for Tulips1 database, while Figure 6 demonstrates a sample image from M2VTS database.

We focused in our experiments mainly on the Tulips1 database, because in these images, the contrast between lips and skin regions is lower as compared to M2VTS, so the correct extraction of the lip contour for the images in Tulips1 is more difficult.

With the very "primitive" initialization of the heuristic edge following algorithm proposed in this work, the algorithm can be prone to instability. The final result depends on the choice of the starting point of each region, performed by the human, to a high degree. In a future work the first improvement of the algorithm will address this problem by including some a-priori information about the hue patterns in each region of the lip contour in the initial direction value considered as first reference in the search.

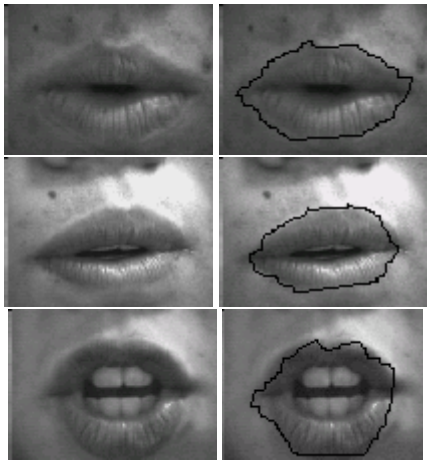

Figure 5. Three outer lip contour extraction examples, for 3 subjects from Tulips1 database

**Figure 6. Outer lip contour extraction example for M2VTS database.
From left to right: the original mouth image; outer lip contour tracked with our algorithm; outer lip contour tracked with B-splines**

5. Conclusions

We developed a new solution to the problem of lip contour extraction in gray-level images based on image gradient information. The proposed approach performs contour detection using the Sobel edge detector and heuristic edge following. Compared to other similar algorithms, the solution proposed here has the advantage of providing a reliable lip contour without any geometric model assumption, while requiring a small number of points (6 to 12) to be manually labeled. In a future work, the algorithm will be made completely automatic, by learning the directional patterns from a set of training mouth images and using the learned patterns to initialize edge following.

6. Acknowledgement

This work was supported by the European Union Research Training Network "Multi-modal Human-Computer Interaction (HPRN-CT-2000-00111)".

References

[1] R. Kaucic, B. Dalton, and A. Blake. "Real-time lip tracking for audio-visual speech recognition applications," in *Proc. European Conf. Computer Vision*, Cambridge, UK, 1996, pp. 376-387.

[2] S. Dupont and J. Luettin. "Audio-visual speech modeling for continuous speech recognition," *IEEE Transactions on Multimedia*, 2(3):141-151, Sept. 2000

[3] I. Matthews, T. Cootes, S. Cox, R. Harvey, and J. A. Bangham. "Lipreading Using Shape, Shading and Scale," in *Proc. Auditory-Visual Speech Processing*, Sydney, Australia, December 1998, pp. 73-78.

[4] J. Luettin, N. A. Thacker, and S. W. Beet. "Active shape models for visual speech feature extraction," in *Speechreading by Humans and Machine, NATO ASI Series, Series F: Computer and Systems Sciences*, 150:383-390, Springer Verlag, Berlin, 1996

[5] M. U. Ramos Sanchez, J. Matas, and J. Kittler. "Statistical chromaticity models for lip tracking with B-splines," in *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication*, Crans Montana, Switzerland, 1997, pp. 69-76.

[6] J. Luettin and N. A. Thacker. "Speechreading using probabilistic models," *Computer Vision and Image Understanding*, 65(2):163-178, February 1997

[7] M. E. Hennecke, K. V. Prasad, and D. G. Stork. "Using deformable templates to infer visual speech dynamics," in *Proc. 28th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, November 1994, pp. 578-582.

[8] Y. Tian, T. Kanade, and J. F. Cohn. "Robust lip tracking by combining shape, color and motion," in *Proc. of ACCV'2000*, Taipei, Taiwan, January 2000, pp. 1040-1045.

[9] I. Pitas. *Digital Image Processing: Algorithms and Applications*. John Wiley & Sons, Inc., February 2000

[10] *http://www.ee.surrey.ac.uk/EE/VSSP/xm2vtsdb/results/lips/*

[11] J. R. Movellan. "Visual Speech Recognition with Stochastic Networks," in *Advances in Neural Information Processing Systems,* (G. Tesauro, D. Toruetzky, and T. Leen, Eds.), Vol 7, MIT Pess, Cambridge, MA, 1995

[12] S. Pigeon and L. Vandendorpe. "The M2VTS multimodal face database," in *Lecture Notes in Computer Science: Audio- and Video- based Biometric Person Authentication* (J. Bigun, C. Chollet and G. Borgefors, Eds.), vol. 1206, pp. 403-409, 1997