# LevelSet versus AGSnakes as Mouth's Shape Extraction Algorithm

ETIENNE BOUAN, ARNAUD CHARDON, DAVID MARTÍNEZ,
NICOLAS CLADEL, RENAUD SÉGUIER
SCEE
SUPÉLEC
Avenue de la Boulaie, 35511 Cesson-Sévigné
FRANCE
http://www.rennes.supelec.fr/ren/rd/scee/

*Abstract:* - This paper analyses the performance of a proposed mouth shape's extraction algorithm based on LevelSet while comparing it with an existing alternative: the MultiObjective Genetic Snakes. This new algorithm offers an arbitrary initialisation and is 150 times faster than the previous one. Its implementation and theoretical bases are described.

*Key-Words:* - LevelSet, Snakes, Lipreading, Human Computer Interaction.

## 1 Introduction

Our interest focus in speech recognition and especially in the help that lips reading can bring to this recognition. Indeed today's technologies allows the use of visual information in addition to the audio part in speech recognition. Thanks to this additional information it becomes possible to perform recognition under very noisy conditions.

Finding the lips on a picture requires first to isolate the face and a region of interest around the mouth. Then several techniques allows to recover the outline of the lips but most of them (shape model [14], dynamic contours [15], deformable models [16]) require an initialisation close to the goal.

We compare here two algorithms which allow the use of an arbitrary initialisation: MultiObjective Genetic Snakes (MOGS) [3] and a LevelSet approach [2].

Genetic algorithms are commonly used in optimisation problems for their ability to avoid local minima. As well as this, MOGS allows both to overcome the problem of snakes initialisation and not to weight the different energies minimised in the classical snakes as they are considered in parallel in a multiobjective framework.

We propose here a LevelSet approach which also allows an arbitrary initialisation. With LevelSet we can recover an outline without any prior knowledge of its shape.

Both techniques use a preprocessing algorithm described in section 2 which isolates the region of interest and in this region the lips colour. Sections 3 and 4 present the MOGS and LevelSet approaches. The two techniques have then been tested on the European database M2VTS [1]. The results of these test are given in section 5 and we then bring out advantages and drawback of both in section 6.

## 2 Pre-Processing

Both algorithms can't directly work on colour images, as the color tone between the lips and the skin are too similar. In order to resolve this issue, a preprocessing to reveal the lips and the mouth from the rest of the face is required.

The face and the line Lmouth are first located using the approach described in [3]. In the V values (from YUV colour coordinate system), the lips have a strong level of intensity while the teeth and the dark interior of the mouth are confused and rather dark (Fig.1b). On the first images, the RGB signature of the pixels belonging to the lips (those which have a height V value around Lmouth) is evaluated. It is also memorized the signature of those which belong to the skin (black areas in Fig.1a). Then the pixels are classified by evaluating the Euclidean distance between the RGB value of the pixel and each of the two signatures (Fig.1c). The gravity centre $C_G$ of the interior of the mouth is then computed starting from the white pixels of Fig.1c. The edges are finally extracted from this last image.

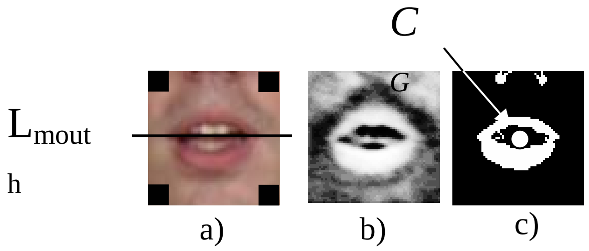

Fig.1. Mouth Preprocessing.

The gravity centre $C_G$ will help to initialise the first snakes' points.

## 3 AgSnakes Algorithm

AGSnakes [3] use the optimisations given by the Genetic algorithms and Genetic Snakes [11][12][13]. Genetic algorithms help to avoid local minima [10]. Merged with the classical snakes, they enables not to initialise the contours near the lips to identify them.

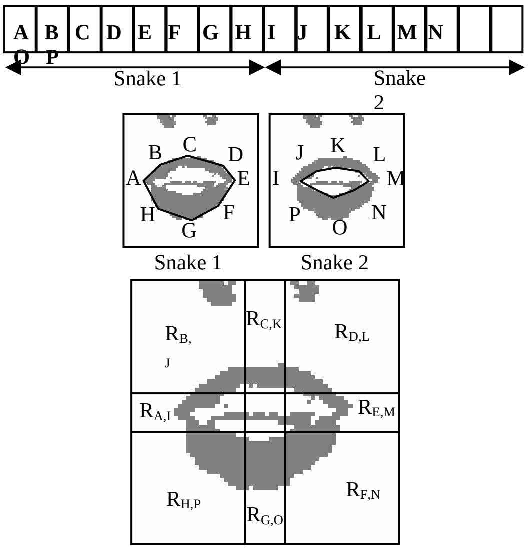

Fig.2. AGSnakes' mouth segmentation for two eight nodes snakes.

After obtaining the mouth segmented image in the preprocessing, two snakes will be used to identify two zones. The first one is composed by the mouth's interior which can be seen only when the mouth is open. The second one is composed of the lips. Each shape has 32 points distributed around the gravity centre $C_G$, found during the preprocessing. These zones limit the choice of the initial nodes' position for each node of the snake. One node is chosen randomly in one zone delimited from the gravity centre. For example the region $R_{B,J}$ in Fig. 2 is dedicated to the nodes *B* and *J* of the snakes 1 and 2.

With all the nodes chosen, the chromosomes can be formed. A chromosome is composed by 32 or 64 genes depending on the numbers of snakes used. Each gene represents the coordinate of each node (Fig. 2).

The AGSnakes minimise five energies. Two types of energies are used on both snakes and another one on the chromosome. The first energy used the numbers of pixels which belong to the object to identify and those who are not. This energy is the difference between these two numbers. When minimizing, the contour grows to include new good pixels even if it absorbs wrong pixels. The second energy uses the variance of the pixels contained by the snakes. When minimizing, this energy tends to reduce the contour to obtain a homogeneous surface. The third energy is a gradient energy; it evaluates the gradient on the contour's points. It makes the snakes to tick to the pixel with an high value of gradient.

In a multiobjective framework, these energies are minimised in parallel. A system which will use a weighted sum of these energies will lead to a less robust lips modelisation [3].

## 4 LevelSet's Algorithm

### 4.1 LevelSet

Invented in the late 80's by Osher and Sethian [2], LevelSets are a new way to describe a curve, through an implicit function $\phi$, the curve being the zero level of $\phi$. Using this implicit function allows the curve to split, rejoin or disappear: the description does not depend on the topologic structure of the curve.

This description can easily be used for image processing: an initial curve is set as well as a rule of evolution for $\phi$. This rule is supposed to make the curve move to the wanted outline, as for snakes. Several rules have been successfully proposed [6], which has made LevelSets famous.

The problem with LevelSets is that dealing with $\phi$ can become very time consuming. This is why optimisations (and simplifications) have been proposed: we can quote the Narrow Band [4] technique and the Fast Marching [5] one. The later allows a considerable speed improvement but requires strong constraints for initialisation: initial curve must be either totally inside or totally outside of the wanted outline since it can only move in one direction.

Edges are not very pronounced in our application. So we chose the 'region driven' [7][8] approach which mainly uses the colour of the picture and not its gradient.

This type of algorithm tries to divide the picture into two parts (inside and outside) with an homogeneous colour in each part. So it searches the curve C which minimizes both

$$\int_{in(C)} \left|u_0(x,y) - c_1\right|^2 dxdy \quad \text{and} \quad \int_{out(C)} \left|u_0(x,y) - c_2\right|^2 dxdy$$

where $u_0$ is the picture and $c_1$ and $c_2$ are the average colours of the inside and outside parts. An extra term is added to penalise too long outlines. So the energy to minimise is:

$$\begin{aligned} E(C,c_1,c_2) &= \mu \cdot Length(C) + \\ &\lambda_1 \cdot \int_{in(C)} \left|u_0(x,y) - c_1\right|^2 dxdy + \\ &\lambda_2 \cdot \int_{out(C)} \left|u_0(x,y) - c_2\right|^2 dxdy \end{aligned} \qquad (1)$$

This formulation can be transposed for LevelSets [7] and gives the following rule of evolution for the implicite function:

$$\frac{\partial\phi}{\partial t} = \delta(\phi)\left[\mu \cdot div\left(\frac{\nabla\phi}{\nabla|\phi|}\right) - \lambda_1(u_0 - c_1)^2 + \lambda_2(u_0 - c_2)^2\right]$$
(2)

$\delta$ is the first derivative of the Heaviside function. In practice it is approximated by a continuous function. $c_1$ and $c_2$ should be re-evaluated after each iteration. In our application, the pre-processsing gives a binary picture so we already know the colour of what should be the inside and the outside part. This is why we fixed $c_1$ and $c_2$ from the beginning. Since the algorithm has this information, the wanted outline can be found even if it is not close to the initial curve. So we can place this initial curve anywhere on the picture. We chose a circle-grid initialisation [9]. The small circles will disappear if they are outside and rejoin if they are inside as we can see in Fig. 3 on a medical example. As any point on the picture is not very far from the initial curve, convergence is very quick.

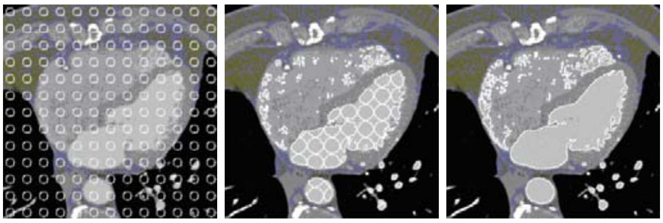

Fig. 3 (a) initial position, (b) after one iteration, (c) after 5 iterations.

On this example we see that the wanted outline around the light area is found with good precision but other parasitic outlines are found elsewhere. This is a general result for this algorithm. So a post processing is necessary to select one or several pertinent outlines.

Fig. 4a shows the initialisation on a mouth example. On this type of picture this method allows to completely detect all objects in less than three iterations. It is robust and efficient, but not necessarily optimum. In fact it does not use the fact that in lipreading applications the target's shape is known and could be easily modelled. Tests also show that a double concentrically circle initialisation (Fig. 4b) gives better results in some cases while reducing the amount of parasites, and allow to detect better the internal shape if the convergence speed is controlled.

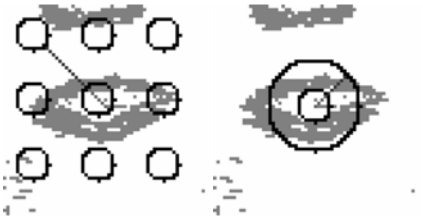

Fig. 4. (a) Grid Initialisation, (b) Double concentrical initialisation.

The preprocessing described bellow which produces a picture with two colours dedicated for the lips and the skin is not perfect: the result can be very noisy as on Fig.5a. To avoid this noise we used a filter applied to the variation of $\phi$. So between two iterations we calculate $\phi_{n+1} = \phi_n + G \cdot \Delta\phi$ instead of $\phi_{n+1} = \phi_n + \Delta\phi$, where G is a low-pass filter.

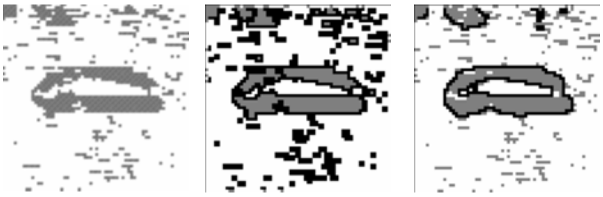

Fig.5 (a) Picture after preprocessing, (b) result without filter (c) result with filter.

## 4.2 LevelSet's Post-Processing

At the end the mouth can be recovered as one outline (closed mouth, as on Fig. 6) or two either separated (last image in Fig. 9) or with one included in the other (inside and outside outlines of the lips when the mouth is open, Fig. 7). The post-processing allows to calculate the external outline and in some cases, even the internal one.

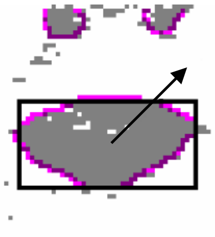

Fig. 6. A closed mouth.

The algorithm consists in starting at the centre of the picture and moving along a diagonal until we find an outline given by the LevelSet (so a zero of ϕ) as in Fig.6. Then we follow this outline and calculate its height and width. We do this for two outlines found when moving to the top and one to the bottom if we had found twice the same. At the end we have two different outlines maximum which have been found either both when moving to the top or one in each side. According as they

are imbricate or not we consider them as the inside and outside outline of the mouth or as the two lips.
More difficult cases can occur (with parasitic outlines or an outline touching the edge of the picture as we will see in section 5). In those cases the postprocessing algorithm is less reliable.

## 5 LevelSet's Test Results

Our algorithm was tested on 6000 images of the European Data Base M2VTS (Multi Modal Checking for Teleservices and Security applications [1]). This base is dedicated to audio-visual recognition and identification. The persons pronounces four times (at one week interval) the digits from 0 to 9. The images were acquired at 25Hz with a weak resolution (288x360 pixels in 4:2:2). These images were previously used for testing an AGSnakes based lipreading algorithm. Total running time was of 0.3s by image in Matlab.

For the lipreading module, only the external shape of the mouth is used. Results can then be classified in the following categories:

- *(A) Excellent* Both internal and external shape of the mouth are detected (Fig.7).
- *(B) Success* Only the external shape is perfectly detected (Fig.8).
- *(C) Partial-Success* The external shape is approximately detected (Fig.9).
- *(F) Miss* The external shape is not correctly detected (Fig.10abc).

One or two rectangles are drawn when the algorithm has detect the external or both shapes of the lips.

**A-type** The LevelSet algorithm is able to provide this type of results in the *45%* of the test images. A-type are obtained when the preprocessing module output consist of a low-noise image.

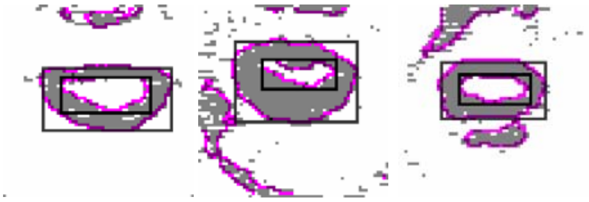

Fig. 7. A-type images. Both shapes are perfectly detected.

**B-type** As consequence of the lack of pre-modelling capabilities in the proposed algorithm, the output of the LevelSet segmentation could not provide the post-processing module a continuous shape of the mouth (left image in Fig.8) and then the continuous internal shape is lost.

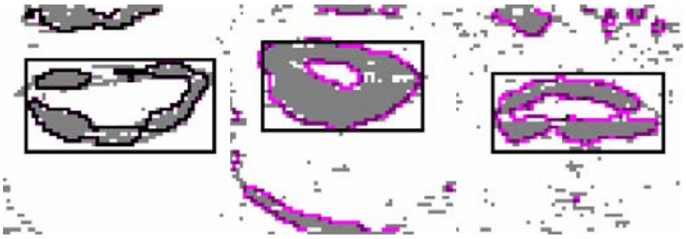

Fig 8. B-type images. Only the external shape is perfectly detected.

The proposed post-processing algorithm is simple and it considers that the centre of the image is placed within the mouth. If that condition is not satisfied (central image in Fig. 8), the internal shape is not well labelised. Nevertheless, lipreading algorithm could use just the external shape to work and then these results are considered as successful. The algorithm provides a *18%* of B-type results.

**C-type** As presented in the previous section a filter is used to reduce the image noise. Sometimes, as in B-type images, the segmentation doesn't give a continuous shape. If the discontinuity is enough to completely separate the lips then the post-processing algorithm will correctly detect the height but not the width (right and left images in Fig. 9). If the error is small the image will be considered C-type, elsewhere it will be classified F.

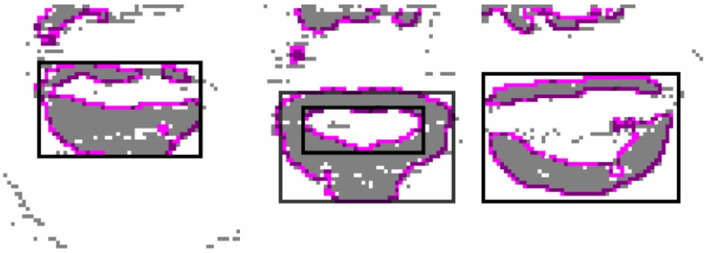

Fig. 9. C-type images. External shape is detected. Small error in either the length or the width.

Modifying the filter could reduce this problem, but in the other hand parasites could be interpreted to be part of the mouth, hence introducing an error in the measure. The most important parasite is the chin (central image of Fig.9). If this error remains constant all over the sequence, it doesn’t decrease the lipreading performances (height and width dynamics are used by the classification module). *14%* of the test images have been classified as C-type.

A, B and C types represent *77%* of the test set.

**F-type** In *23%* of the test set (Figs. 10a, b, c) the algorithm is not able to find the mouth shape. This is cause by either a highly noised source image (Fig. 10b) or a post-processing mistake as it founds ‘holes’ in the lips (Fig. 10a, c). As we can see in Fig. 10def on the same images, the AGSnakes are sometimes able to find interior height and width of the mouth even if the lips have ‘holes’ but have difficulties to find the correct shape if there is too much noise in the segmented image (Fig10e).

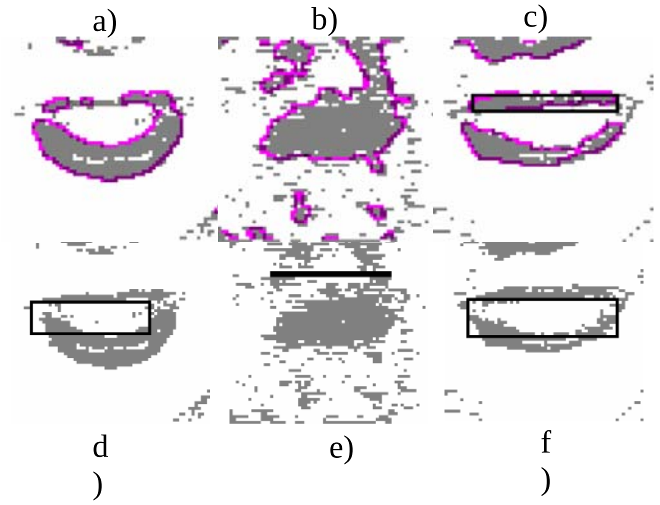

Fig. 10 F-type images. No external shape detected. First line LevelSet's results, second line AGSnakes's ones.

## 6 LevelsSet vs AGSnakes Performance Comparison

LevelSet algorithm is faster (0.3s vs 50s with 64x64 under Matlab) than the AGSnakes. This is mainly due to the iterative evaluation of the energy which is minimise in the LevelSet proposition. In comparison, the differents energies considered by the AGSnakes must be completely evaluated over the different snakes (which constitute the population) at each Genetic Algorithm iteration.

Nevertheless, one of the greatest issues while using LevelSets concerns the initialisation. For both algorithm, we only need to know the centre of the mouth as a priori information. But the exploitation of this information is more heavy in AGSnakes (as describe in section 3) and lead to a more time consuming implementation.

In contrast with the AGSnakes solution where its multi-objective nature allows it to introduce more a-priori information like a mouth model, a multi-objective adaptation of the LevelSet is rather difficult and leads to a more difficult consideration of an a priori mouth model.

Table I summarise the comparison between LevelSet and AGSnakes propositions.

| Parameter | LevelSets | AGSnakes |
|---|---|---|
| Type | Mono-objective | Multi-objective |
| Information | Local | Global and Local |
| Converg. speed | High | Low |
| Complexity | Low | High |
| Precision | Medium | High |
| A-priori model. | No | Yes |
| Initialization | Not required | Not required |

Table I. Global comparison between AGSnakes and LevelSets.

Table II shows a comparison between the quality of the LevelSets algorithm and the previous AGSnakes' solution in the lipreading context.

As seen, the LevelSets algorithm offers a near real-time speed with a quality comparable with the AGSnakes solution if the lipreading algorithm can use only the characteristics (height and width) of the external shape of the mouth, which is the case here. Nevertheless it offers a lightly worse noise tolerance .

| Parameter | LevelSets | AGSnakes |
|---|---|---|
| Speed | *0.3s/image* | *50s/image* |
| Internal H res.<br>Internal V res. | -<br>- | +<br>+ |
| External H res.<br>External V res. | +<br>++ | ++<br>++ |
| Noise Tol. | ++ | +++ |

Table II. Quality comparison between AGSnakes and LevelSets.

## 7 Conclusion

In this article we have presented an alternative to the AGSnakes algorithm in mouth's shape extraction for lipreading applications. It is constituted by a segmentation algorithm based on LevelSet and a post-processing module to extract the correct shape.

If the lips and skin colours are rather different, it is possible to produce segmented images of the lips which are not to noisy. In that case, the proposed LevelSet algorithm allows to achieve a quality similar in the external shape to the AGSnakes option with a much lower complexity that allows to improve the convergence speed by a factor of 150.

In conclusion, even if the post-processing algorithm gives satisfactory results and is simple, it could be enhanced in future researches.

*References:*

[1] S. Pigeon, M2VTS, http://www.tele.ucl.ac.be/PROJECTS/M2VTS/m2fdb.html, 1996.

[2] Osher, Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computational* Physics 79, 12-49, 1988.

[3] R. Séguier, Nicolas Cladel, Multiobjectives genetic snakes: application on audio-visual speech recognition, *EURASIP*, 2003.

[4] D. Adalsteinsson, J.A. sethian, A Fast LevelSet Method for Propagating Interfaces, *Journal of computational Physics*, 118(2):269-277, 1995.

[5] J.A. Sethian, A Fast Marching LevelSet Method for Monotonically Advancing Fronts, *Proc Nat. C. science*, volume 93, pages 1591-1694, 1996.

[6] R.Malladi, J.A. Sethian, B. Vernuri, Shape modelling with front propagation: A level set approach, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(2):158-174, 1995.

[7] T.Chan and L.Vese, Active contours without edges, *IEEE Trans. Image Processing*, 10(2):266-277, February 2001.

[8] M. Wasilewski, Active Contours using Level Sets for Medical Segmentation, *http://www.cgl.uwaterloo.ca/ ~mmwasile/cs870/*

[9] T.Chan and L.Vese, A new multiphase LevelSet framework for image segmentation via the Munford and Shah model. *Tech. report CAM. report 01-25.UCLA.* April 2001.

[10] K. Sakaue and A. Amano and N Yokoya, Optimization approaches in computer vision and image processing, *IEICE Trans. Inf. and Syst.*, 1999.

[11] A. Cagnoni and A. Dobrzeniecki and R. Poli and J. Yanch, Genetic algorithm-based interactive segmentation of 3D medical images, *Image and Vision Computing*, 17(12): 881- 895, 1999.

[12] L. Ballerini, Genetic snakes for color images segmentation, *Lecture Notes in computer sciences*, 2037, 2001.

[13] N. Covavisaruch and T. Tanatipanond, Deformable Contour for Brain MR Images by Genetic Algorithm: From Rigid toTraining Approaches, *Proceedings Image and Vision Computing New Zealand (IVCNZ)*, 1999.

[14] Michael T. Chan, You Zhang, and Thomas S. Huang, Real-time lip tracking and bimodal continuous speech recognition, *Workshop on Multimedia Signal Processing*, 1998.

[15] P. Delmas, P.Y. Coulon, and V. Fristot. Automatic snakes for robust lip boundaries extraction, *International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 1999.

[16] Y. Tian, T. Kanade, and J. F. Cohn, Recognizing action units for facial expression analysis, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2001.