

Shape-Based Active Contours for Fast Video Segmentation

Sasan Mahmoodi

Abstract—In this letter, we propose a shape-based active contours method for segmentation, based on a piecewise-constant approximation of the Mumford–Shah (M–S) functional. The Chan–Vese (C–V) formalism in a level set framework is used to formulate our method; however no sign distance function (SDF) is employed in the method proposed here. This method has the topology-free segmentation associated with the C–V algorithm and adds faster convergence, less memory requirement and fast re-initialization. These properties make the algorithm very attractive for video segmentation.

Index Terms—Active contours, Chan–Vese model, image, Mumford–Shah functional, segmentation, shape characteristic function, video.

I. INTRODUCTION

THERE has been a great deal of interest in recent years in the level set method for image segmentation. In this framework, contours are represented implicitly as a zero level of a SDF [1]. A contour evolution is then achieved by using gradient descent methods to implement Euler–Lagrange equations minimizing an appropriate functional with respect to SDFs. Shapes, on the other hand, have become popular in a prior shape segmentation framework because of the favourable properties of their dissimilarity measure (see e.g., [2]). In this paper, we are motivated by the work of Cremers *et al.* [2] to propose a framework in which shapes represented by their characteristic functions instead of SDFs, are considered as variables whose evolution leads to segmentation. We demonstrate here that the pioneering work of the Chan *et al.* proposed in [3] leads to a framework in which SDFs are not required. We note that the existence of a shape-based segmentation method is indicated in [3]; however to the best of our knowledge it has not been explored in the literature. Our contribution in this paper is therefore to employ C–V formalism of S–M model [5] using shape characteristic functions as variables to develop a segmentation method enjoying the same properties associated with the level set method without using SDFs. The advantages of such a framework are

- 1) there is only one type of variable (shape characteristic function) and there is therefore no need to switch between variables (SDFs, and their Heaviside functions) in an algorithm;

Manuscript received March 24, 2009; revised June 10, 2009. First published June 23, 2009; current version published July 17, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Dimitrios Androustos.

S. Mahmoodi is with the School of Electronic and Computer Science, Southampton University, Southampton, SO17 1BJ, U.K. (e-mail: sm3@ecs.soton.ac.uk).

Digital Object Identifier 10.1109/LSP.2009.2025924

- 2) less memory is used to implement the method proposed here;
- 3) the proposed algorithm converges to the final solution faster than a level set-based algorithm;
- 4) the re-initialization of the shape characteristic function is fast and computationally much less expensive than SDF re-initialization;
- 5) The method proposed here has a better performance at the presence of noise.

Our paper consists of five sections: a brief description of C–V model and the modification we propose are discussed in Section II; Section III deals with the implementation issues; experimental results are presented in Section IV; and finally, conclusions are in Section V.

II. FORMULATION

The level set based functional proposed in [3] is a piecewise constant approximation of M–S functional, and consists of three terms:

$$E(c_1, c_2, \phi) = \iint_{\Omega} \{ \mu |\nabla H(\phi(x, y))| + \lambda |g(x, y) - c_1|^2 H(\phi(x, y)) + \nu |g(x, y) - c_2|^2 (1 - H(\phi(x, y))) \} dx dy. \quad (1)$$

In functional (1), the first term minimizes the contour length, the second and third terms are responsible for data fidelity. $g, \phi, \nabla, H, \Omega, c_1$ and c_2 are the image data, SDF, gradient operator, Heaviside function, image domain, the mean grey scale inside and outside of the contour respectively. Coefficients μ, λ and ν are constant associating a weight to each term. The associated Euler–Lagrange equation for $\phi : \Omega \rightarrow R$ is then derived [3]:

$$\frac{\partial \phi}{\partial t} = \delta(\phi) \left(\mu \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|} - \lambda |g - c_1|^2 + \nu |g - c_2|^2 \right) \\ \phi(0, x, y) = \phi_0(x, y) \text{ in } \Omega \quad \text{and} \quad \frac{\delta(\phi)}{|\nabla \phi|} \frac{\partial \phi}{\partial \vec{n}} = 0 \text{ on } \partial \Omega \quad (2)$$

where \vec{n} denotes the exterior normal to the boundary $\partial \Omega$. In this paper, we rewrite functional (1) using shape characteristic function $\chi : \Omega \rightarrow [0, 1]$ instead of ϕ :

$$E(c_1, c_2, \chi) = \iint_{\Omega} \{ \mu |\nabla \chi| + \lambda |g(x, y) - c_1|^2 \chi(x, y) + \nu |g(x, y) - c_2|^2 (1 - \chi(x, y)) \} dx dy. \quad (3)$$

Accordingly one can define inside and outside of a contour C surrounding the shape with characteristic function χ as

$$\begin{aligned} \text{inside}(C) &= \{(x, y) \in \Omega \mid \chi(x, y) = 1\}, \\ \text{outside}(C) &= \{(x, y) \in \Omega \mid \chi(x, y) = 0\}. \end{aligned}$$

We therefore arrive at Euler-Lagrange equation:

$$\begin{aligned} \frac{\partial \chi}{\partial t} &= \mu \nabla \cdot \frac{\nabla \chi}{|\nabla \chi|} - \lambda |g - c_1|^2 + \nu |g - c_2|^2 \\ \chi(0, x, y) &= \chi_0(x, y) \text{ in } \Omega \quad \text{and} \quad \frac{1}{|\nabla \chi|} \frac{\partial \chi}{\partial \vec{n}} = 0 \text{ on } \partial \Omega \end{aligned} \quad (4)$$

where \vec{n} denotes the exterior normal to the boundary $\partial \Omega$. It is noted that a Dirac delta function is not present in (4), since the minimization is performed with respect to χ rather than SDF. This is the main reason for higher speed of convergence of the present method.

The presence of $\delta(\phi)$ in the right hand side of (2) causes ϕ to be updated over the domain where $\delta(\phi)$ is not close to zero. Therefore changes in ϕ anywhere else in the image domain are not significant. Since the contour evolves iteratively, updates in ϕ propagate in the regions where the contour evolution has occurred until convergence. However this is not the case for (4). Function χ is updated everywhere in each iteration causing the present method to converge faster to the solution. Minimization of (3) with respect to c_1 and c_2 results in

$$c_1(\chi) = \frac{\iint_{\Omega} g(x, y) \chi(x, y) dx dy}{\iint_{\Omega} \chi(x, y) dx dy} \quad (5)$$

$$c_2(\chi) = \frac{\iint_{\Omega} g(x, y) (1 - \chi(x, y)) dx dy}{\iint_{\Omega} (1 - \chi(x, y)) dx dy}. \quad (6)$$

III. IMPLEMENTATION ISSUES

To implement the shape based active contour described in Section II, we need to find how to use shape characteristic functions in a segmentation algorithm. The first problem is how to generate or reconstruct a shape characteristic function without using any SDF. The second is that how such a characteristic function is regularized. The third is that how the evolving contour is represented in a segmentation framework based on shape characteristic functions. To answer these questions, we note the following key observation:

$$H\left(\chi(x, y) - \frac{1}{2}\right) = \chi(x, y) \quad (7)$$

where H is the Heaviside function, i.e.,

$$H(x) = \begin{cases} 1, & x > 0 \\ 0, & x < 0 \end{cases}$$

Equation (7) can be employed to generate or reconstruct the shape characteristic function to solve the first problem. In our implementation, we start with an arbitrary initial non smooth shape characteristic function $\chi_0^0: \Omega \rightarrow \{0, 1\}$ where in χ_0^0 superscript and subscript represent the iteration number and regularizing parameter (a zero subscript indicates that the initial shape is not regularized) respectively. We then use (8) to regularize χ_0^0 solving the second problem, i.e.,

$$\chi_{\varepsilon}^0(x, y) = G_{\varepsilon}(x, y) * \chi_0^0 \quad (8)$$

where $G_{\varepsilon}(x, y) = (\varepsilon)/(\pi(x^2 + y^2 + \varepsilon^2))$ and ε are a Dirac filter (regularized Dirac delta) used to construct C^{∞} regularized shape characteristic function and regularizing parameter. Our rationale to use (8) to regularize the shape characteristic function is that as $\varepsilon \rightarrow 0$, $G_{\varepsilon}(x, y)$ converges to $\delta(x, y)$ Dirac delta function. Also at iteration t , (7) is used to reconstruct χ_0^t , i.e.,

$$\chi_0^t(x, y) = H\left(\chi_{c\varepsilon}^t(x, y) - \frac{1}{2}\right) \quad (9)$$

where t and $\chi_{c\varepsilon}^t$ are the iteration number and the shape characteristic function calculated by solving PDE(4).

Equation (8) is then exploited to regularize the characteristic function for the current iteration, i.e.:

$$\chi_{\varepsilon}^t(x, y) = G_{\varepsilon}(x, y) * \chi_0^t \quad (10)$$

Function χ_{ε}^t calculated in (10) should have a range in interval $[0, 1]$. It is therefore normalized to calculate $\chi_{n\varepsilon}^t$ to ensure that its minimum and maximum correspond to zero and one respectively. The subscript n in $\chi_{n\varepsilon}^t$ indicates that the reconstructed χ is normalized. The following analytical argument also suggests a solution for the third problem. In a segmentation algorithm based on SDFs, the shape characteristic function associated with $\phi(x, y)$ is calculated by using a regularization method proposed in [3] as

$$\chi(x, y) = H(\phi(x, y)) = \frac{1}{\pi} \arctan \frac{\phi(x, y)}{\eta} + \frac{1}{2}$$

where η is the regularizing parameter and $H(\cdot)$ is the Heaviside function.

It is clear from the above equation that the evolving contour represented as the zero level of SDF, i.e., $\phi(x, y) = 0$, corresponds to $\chi(x, y) = \frac{1}{2}$. In a segmentation scheme based on shape characteristic functions, the evolving contour at iteration $t + 1$ is therefore detected as the zero-crossing of $\chi_{n\varepsilon}^{t+1} - \frac{1}{2}$. Equation (4) is finally discretized by using semi-implicit finite difference method [4] to calculate $\chi_{c\varepsilon}^{t+1}$ for iteration $t + 1$ by regarding $\chi_{n\varepsilon}^t$ as the shape characteristic function at iteration t . It is also noted that we use only one variable (χ) to calculate the evolving contour. This is in contrast with the C-V algorithm where two variables (SDF and its shape characteristic function) are required in each iteration. If the input image is $M \times N$, the algorithm proposed here requires less memory (reduced by $M \times N$ double precision numbers) for implementation. The segmentation algorithm for a single frame image therefore consists of the following steps.

1. Initialize χ_0^0 for $t = 0$.
2. construct χ_{ε}^t using (9) and (10) and then normalize it to calculate $\chi_{n\varepsilon}^t$.
3. Detect the zero-crossing of $\chi_{n\varepsilon}^t - \frac{1}{2}$ as the evolving contour at iteration t .
4. Compute c_1 and c_2 using (5) and (6).
5. Solve PDE (4) with respect to χ to calculate $\chi_{c\varepsilon}^{t+1}$.
6. Check whether a convergence is reached, if not, go to step 2.

It is noted that re-initialization for the shape characteristic function using (9) and (10) is fast and numerically less expensive than re-initialization for a signed distance function. This is an important feature for a fast video segmentation. Using the algorithm proposed here, one can use the final shape characteristic function obtained from the segmentation of a frame as an initial shape characteristic function χ_0^0 for the next frame.

It should be noted that this is not feasible in the C-V algorithm based on SDFs, because the final SDF obtained from the

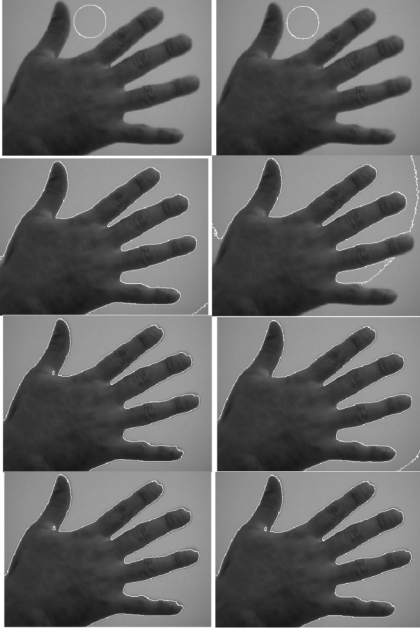


Fig. 1. Contour evolution for our algorithm and C-V algorithm: Our algorithm (left column from top to bottom): Initial contour, Second iteration, Third iteration, Fourth iteration; C-V algorithm (right column from top to bottom): Initial contour, 1000th iteration, 4000th iteration, 8500th iteration.

segmentation of a frame is usually too flat to be used as an initial SDF for the next frame. Therefore a re-initialization for SDF is required before the segmentation process is started for the next frame. The re-initialization of a SDF can be accomplished by Fast Marching algorithm [1] or by solving the following eikonal equation [6]:

$$\begin{aligned} \frac{\partial \phi}{\partial t} &= \text{sign}(\varphi_f^{F-1}) (1 - |\nabla \phi|) \\ \phi(0, x, y) &= \varphi_f^{F-1} \end{aligned} \quad (11)$$

where φ_f^{F-1} , t and sign are the final SDF obtained by segmenting the object of interest in frame $F - 1$, the artificial time and sign function respectively. The steady state solution of (11) results in a re-initialized SDF $\phi(x, y)$ being the initial SDF for the next frame. Both of the above methods for SDF re-initialization are numerically more expensive than (9) and (10) for the initialization of shape characteristic function.

IV. EXPERIMENTAL RESULTS

In the first example, a 288×388 real world image of a hand shown in Fig. 1 is used for segmentation.

The initial contours are shown in the right column of Fig. 1 for both the method proposed here and the C-V algorithm. The algorithm proposed here converges to the solution in four iterations (the left column in Fig. 1 from top to bottom). However, it takes 8500 iterations for the algorithm proposed in [3] to converge to the final solution. It is noted that the same parameters with the same initial contour are used in both algorithms, ($\Delta t = \lambda = \nu = \mu = 1$ and the radius of the initial circle is 30). In this paper, same regularization parameter for Dirac delta, is used for both algorithms i.e., $\varepsilon = 1$ for $G_\varepsilon(x, y)$ and for the regularization method proposed in [3]. The size of Dirac filter (regularized Dirac delta) employed in this paper, is set to 5×5 . Some intermediate iterations for contour evolutions in C-V algorithm are shown in the second row of Fig. 1. The

main reason for the fast convergence of the algorithm proposed here in comparison with the C-V algorithm is the absence of the Dirac delta function in (4) causing χ to be updated over the whole image domain in every iteration. Fig. 2 depicts the energy term (1) (or equivalently energy term (3)) calculated in each iteration for both algorithms. As shown in the figure, the convergence of the algorithm proposed here is very fast in comparison with the C-V method. Due to the high speed of convergence, the algorithm proposed here can comfortably be used to segment a sequence of images forming a video. Fig. 3 shows some of the 38 frames of a video taken from a walking person. These 300×400 image frames are contaminated with additive zero mean Gaussian noise. Fig. 4 shows the results obtained by applying the algorithm proposed here on this video sequence with $\Delta t = \lambda = \nu = \mu = \varepsilon = 1$. The level set based algorithm [3] demonstrates similar results to those of the algorithm proposed here but with lower speed. In our algorithm, the final segmentation result (shape characteristic function) of the current frame is always used as the initial shape for segmentation in the next frame. However this is not feasible in the level set-based algorithm in [3], [4], because the SDF becomes flat at the end of the segmentation of the current frame. Therefore the initial SDF for the next frame must be initialized. There are three methods to initialize SDF for the next frame: i) Fast Marching algorithm can generate the SDF associated with the shape segmented in the current frame to be used as an initial SDF for the next frame, ii) The steady state solution of eikonal equation (11) at the end of segmentation of the current frame can be used as an initial SDF for the next frame, iii) an SDF representing a simple ellipse whose centre and main axes are set to be the centre and main axes of the segmented contour of the object of interest in the current frame, is used as an initial SDF for the next frame. The least numerically expensive option (iii) is chosen to implement the fastest method for the level set-based algorithm of [3], [4] for a fair comparison. Fig. 5 shows the time elapsed by CPU to segment the video frames in a PC with a dual core CPU 2.40 GHz in a Matlab version 7.30 environment in two cases: i) when video frames are noise free (Fig. 5(top)) ii) when video frames are contaminated with a zero mean Gaussian noise (Fig. 5(bottom)). For a fair comparison, an error term representing the segmentation accuracy is calculated by computing the absolute difference between manually segmented images as ground truth and the images segmented by the segmentation algorithms investigated here. Such an error term is then divided by the total number of pixels in each frame to obtain the error per pixel.

The CPU time required for the segmentation of video frames with respect to the different values in the error term is depicted in Fig. 5(top). Fig. 5(bottom) shows the CPU time needed for the video segmentation with different levels of noise, when the error term is kept constant at 0.015 per pixel. As shown in Fig. 5, the speed of convergence for the shape-based segmentation method proposed here is higher than the C-V framework with different error terms and various levels of noise. It is also noted that in the favour of the level set-based method, we place the initial contour for the first frame close to the object of interest in that frame. If the initial contour for the first frame was placed somewhere far from the object of interest, the level set based algorithm would require a lot of iterations to converge to the solution (similar to the case depicted in Figs. 1 and 2).

The effect of an arbitrary choice for the initial condition in the level set based method would therefore slow down the segmentation process even further for the whole video sequence.

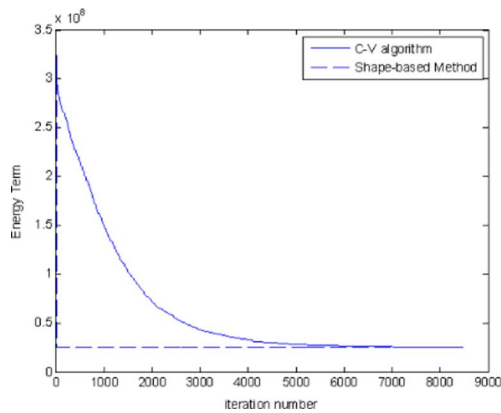


Fig. 2. Energy term calculated from (1) with respect to the iteration number for C-V algorithm (solid line) and the algorithm proposed here (dashed line).



Fig. 3. A few frames of a video sequence of a walking person.



Fig. 4. Segmented results using the shape-based segmentation algorithm proposed here.

For the shape-based method on the other hand, the initial contour for the first frame is already placed in an arbitrary location far from the object of interest. As demonstrated in Fig. 2, the location of the initial contour has no effect in the speed of convergence in the algorithm proposed here. We also notice that the shape-based algorithm presented here shows better performance in the presence of additive zero mean Gaussian noise. While the shape-based algorithm successfully performs the segmentation in noisy frames of Fig. 3 with SNR as low as 1.8, the C-V algorithm starts to segment noise at SNR = 3. This slight improvement in the segmentation performance at the presence of noise in the shape-based method in comparison with C-V algorithm is due to the application of Dirac filter used for regularization.

V. CONCLUSION

It is demonstrated here that the pioneering work of Chan *et al.* to implement Mumford-Shah model for segmentation leads to a framework where no signed distance function is required for implementation. Such a framework has not been explored in the literature.

The contribution of this short paper is to explore this framework and show that the model introduced in [3] can indeed be implemented without any notion associated with the level set theory. Although the present work does not employ any concept related to the level set technique, it enjoys the useful properties

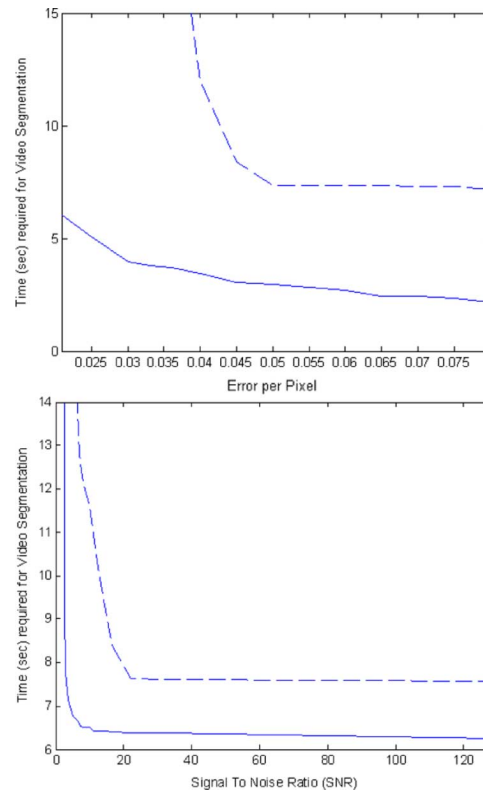


Fig. 5. Time elapsed by CPU to perform the segmentation of a video consisting of 38 frames in the shape-based algorithm proposed here (solid line) and C-V algorithm (dashed line) with respect to (top) error term when the video frames are noise free and (bottom) SNR when the video frames are contaminated with Gaussian noise and the error term is kept constant at 0.015 per pixel.

associated with the original work of Chan *et al.* in [3] such as topology free segmentation. One of the advantages of the work presented here is that it converges to the solution faster than its predecessor. Another advantage is that it uses less memory since it deals with only one variable (shape characteristic function). Last, but not least, the shape-based method shows more robustness at the presence of noise. It is therefore concluded that the shape-based algorithm proposed here is a highly suitable method for video segmentation.

REFERENCES

- [1] J. A. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Material Science*. Cambridge, U.K.: Cambridge University Press, 1996.
- [2] D. Cremers, S. J. Osher, and S. Soatto, "Kernel density estimation and intrinsic alignment for shape priors in level set segmentation," *Int. J. Comput. Vis.*, vol. 69, no. 3, pp. 335–351, 2006.
- [3] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.
- [4] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the Mumford-Shah model," *Int. J. Comput. Vis.*, vol. 50, no. 3, pp. 271–293, 2002.
- [5] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 4, pp. 577–688, 1989.
- [6] M. Sussman, P. Smereka, and S. Osher, "A level set approach for computing solutions to incompressible two-phase flow," *J. Comput. Phys.*, vol. 119, pp. 146–159, 1994.