Fast Geodesic Active Contours

Roman Goldenberg, Ron Kimmel, Ehud Rivlin, Michael Rudzsky

Abstract— **We use an unconditionally stable numerical scheme to implement a fast version of the geodesic active contour model. The proposed scheme is useful for object segmentation in images, like tracking moving objects in a sequence of images. The method is based on the Weickert-Romeney-Viergever AOS scheme. It is applied at small regions, motivated by Adalsteinsson-Sethian level set narrow band approach, and uses Sethian's fast marching method for re-initialization. Experimental results demonstrate the power of the new method for tracking in color movies.**

I. Introduction

An important problem in image analysis is object segmentation. It involves the isolation of a single object from the rest of the image that may include other objects and a background. Here, we focus on boundary detection of one or several objects by a dynamic model known as the 'geodesic active contour' introduced in [4], [5], [6], [7], see also [19], [30].

Geodesic active contours were introduced as a geometric alternative for 'snakes' [32], [18]. Snakes are deformable models that are based on minimizing an energy along a curve. The curve, or snake, deforms its shape so as to minimize an 'internal' and 'external' energies along its boundary. The internal part causes the boundary curve to become smooth, while the external part leads the curve towards the edges of the object in the image.

In [2], [23], a geometric alternative for the snake model was introduced, in which an evolving curve was formulated by the Osher-Sethian level set method [24]. The method works on a fixed grid, usually the image pixels grid, and automatically handles changes in the topology of the evolving contour.

The geodesic active contour model was born latter. It is both a geometric model as well as energy functional minimization. In [4], [5], it was shown that the geodesic active contour model is related to the classical snake model. Actually, a simplified snake model yields the same result as that of a geodesic active contour model, up to an arbitrary constant that depends on the initial parameterization. Unknown constants are an undesirable property in most automated models.

Although the geodesic active contour model has many advantages over the snake, its main drawback is its non-linearity that results in inefficient implementations. For example, explicit Euler schemes for the geodesic active contour limit the numerical step for stability. In order to overcome these speed limitations, a multi-resolution approach was used in [34], and additional heuristic steps were applied in [25], like computationally preferring areas of high energy.

In this paper we introduce a new method that maintains the numerical consistency and makes the geodesic active contour model computationally efficient. The efficiency is achieved by cancelling the limitation on the time step in the numerical scheme, by limiting the computations to a narrow band around the the active contour, and by applying an efficient re-initialization technique.

II. From snakes to geodesic active contours

Snakes were introduced in [18], [32] as an active contour model for boundary segmentation. The model is derived by a variational principle from a non-geometric measure. The model starts from an energy functional that includes 'internal' and 'external' terms that are integrated along a curve.

Let the curve $\mathcal{C}(p) = \{x(p), y(p)\}$, where $p \in [0,1]$ is an arbitrary parameterization. The snake model is defined by the energy functional

$$S[\mathcal{C}] = \int_0^1 \left(|\mathcal{C}_p|^2 + \alpha|\mathcal{C}_{pp}|^2 + 2\beta g(\mathcal{C})\right) dxdy,$$

where $\mathcal{C}_p \equiv \{\partial_p x(p), \partial_p y(p)\}$, and α and β are positive constants.

The last term represents an external energy, where $g()$ is a positive edge indicator function that depends on the image $I(x,y)$, it gets small values along the edges and higher values elsewhere. For example $g(x,y) = 1/(|\nabla I|^2 + 1)$. Taking the variational derivative with respect to the curve, $\delta S[\mathcal{C}]/\delta\mathcal{C}$, we obtain the Euler Lagrange equations

$$-C_{pp} + \alpha\mathcal{C}_{pppp} + \beta\nabla g = 0.$$

One may start with a curve that is close to a significant local minimum of $S[\mathcal{C}]$, and use the Euler Lagrange equations as a gradient descent process that leads the curve to its proper position. Formally, we add a time variable t, and write the gradient descent process as $\partial_t\mathcal{C} = -\delta S[\mathcal{C}]/\delta\mathcal{C}$, or explicitly

$$\frac{d\mathcal{C}}{dt} = \mathcal{C}_{pp} - \alpha\mathcal{C}_{pppp} - \beta\nabla g.$$

The snake model is a linear model, and thus an efficient and powerful tool for object segmentation and edge integration, especially when there is a rough approximation of the boundary location. There is however an undesirable property that characterizes this model. It depends on the parameterization. The model is not geometric.

Motivated by the theory of curve evolution, Caselles et al. [2] and Malladi et al. [23] introduced a geometric flow that includes internal and external geometric measures. Given an initial curve $\mathcal{C}_0$, the geometric flow is given by the planar curve evolution equation $\mathcal{C}_t = g(\mathcal{C})(\kappa - v)\vec{\mathcal{N}}$, where, $\vec{\mathcal{N}}$ is the normal to the curve, $\kappa\vec{\mathcal{N}}$ is the curvature vector, v is an arbitrary constant, and $g()$, as before, is an edge indication scalar function. This is a geometric flow, that is, free of the parameterization. Yet, as long as g does not vanish along the boundary, the curve continues its propagation and may skip its desired location. One remedy, proposed in [23], is a control procedure that monitors the propagation and sets g to zero as the curve gets closer to the edge.

The geodesic active contour model was introduced in [4], [5], [6], [7], see also [19], [30], as a geometric alternative for the snakes. The model is derived from a geometric functional, where the arbitrary parameter p is replaced with a Euclidean arclength $ds = |\mathcal{C}_p|dp$. The functional reads

$$S[\mathcal{C}] = \int_0^1 (\alpha + \tilde{g}(\mathcal{C}))\,|\mathcal{C}_p|dp.$$

It may be shown to be equivalent to the arclength parameterized functional

$$S[\mathcal{C}] = \int_0^{L(\mathcal{C})} \tilde{g}(\mathcal{C})ds + \alpha L(\mathcal{C}),$$

where $L(\mathcal{C})$ is the total Euclidean length of the curve. One may equivalently define $g(x,y) = \tilde{g}(x,y) + \alpha$, in which case

$$S[\mathcal{C}] = \int_0^{L(\mathcal{C})} g(\mathcal{C})ds,$$

i.e. minimization of the modulated arclength $g(\mathcal{C})ds$. The Euler Lagrange equations as a gradient descent process are

$$\frac{d\mathcal{C}}{dt} = \left(g(\mathcal{C})\kappa - \langle \nabla g, \vec{\mathcal{N}}\rangle\right)\vec{\mathcal{N}}.$$

Again, internal and external forces are coupled together, yet this time in a way that leads towards a meaningful minimum, which is the minimum of the functional. [1] One may add an additional force that comes from an area minimization term, and motivated by the balloon force [10]. This way, the contour may be directed to propagate inwards by minimization of the interior. The functional with the additional area term modulated by an edge indicator function reads

$$S[\mathcal{C}] = \int_0^{L(\mathcal{C})} g(\mathcal{C})ds + \alpha \int_\Omega g da,$$

where da is an area element and Ω is the interior of region enclosed by the contour $\mathcal{C}$. The Euler Lagrange as steepest descent, following the development in [37], [36], is

$$\frac{d\mathcal{C}}{dt} = \left(g(\mathcal{C})\kappa - \langle \nabla g, \vec{\mathcal{N}}\rangle - \alpha g(\mathcal{C})\right)\vec{\mathcal{N}}.$$

The connection between classical snakes, and the geodesic active contour model was established in [5] via Maupertuis' Principle of least action [12]. By Fermat's Principle, the final geodesic active contours are geodesics in an isotropic non-homogeneous medium.

[1] An early version of a geometric-variational model, in which $S[\mathcal{C}] = \int g(\mathcal{C})ds / \int ds$, that deals with open curves was proposed in [14].

Recent applications of the geodesic active contours include 3D shape from multiple views, also known as shape from stereo [13], segmentation in 3D movies [21], tracking in 2D movies [25], and refinement of efficient segmentation in 3D medical images [22]. The curve propagation equation is just part of the whole model. Subsequently, the geometric evolution is implemented by the Osher-Sethian level set method [24].

A. Level set method

The Osher-Sethian [24] level set method considers evolving fronts in an implicit form. It is a numerical method that works on a fixed coordinate system and takes care of topological changes of the evolving interface.

Consider the general geometric planar curve evolution

$$\frac{d\mathcal{C}}{dt} = V\vec{\mathcal{N}},$$

where V is any intrinsic quantity, i.e., V does not depend on a specific choice of parameterization. Now, let $\phi(x,y) : \mathbb{R}^2 \to \mathbb{R}$ be an implicit representation of $\mathcal{C}$, such that $\mathcal{C} = \{(x,y) : \phi(x,y) = 0\}$. One example is a distance function from $\mathcal{C}$ defined over the coordinate plane, with negative sign in the interior and positive in the exterior of the closed curve.

The evolution for ϕ such that its zero set tracks the evolving contour is given by

$$\frac{d\phi}{dt} = V|\nabla\phi|.$$

This relation is easily proven by applying the chain rule, and using the fact that the normal of any level set, $\phi = constant$, is given by the gradient of ϕ,

$$\frac{d\phi}{dt} = \langle \nabla\phi, \mathcal{C}_t\rangle = \langle \nabla\phi, V\vec{\mathcal{N}}\rangle = V\left\langle \nabla\phi, \frac{\nabla\phi}{|\nabla\phi|}\right\rangle = V|\nabla\phi|.$$

This formulation enable us to implement curve evolution on the x, y fixed coordinate system. It automatically handles topological changes of the evolving curve. The zero level set may split from a single simple connected curve, into two separate curves.

Specifically, the corresponding geodesic active contour model written in its level set formulation is given by

$$\frac{d\phi}{dt} = \mathrm{div}\left(g(x,y)\frac{\nabla\phi}{|\nabla\phi|}\right)|\nabla\phi|.$$

Including a weighted area minimization term that yields a constant velocity, modulated by the edge indication function, we have

$$\frac{d\phi}{dt} = \left(\alpha g(x,y) + \mathrm{div}\left(g(x,y)\frac{\nabla\phi}{|\nabla\phi|}\right)\right)|\nabla\phi|.$$

We have yet to determine a numerical scheme and an appropriate edge indication function g. An explicit Euler scheme with forward time derivative, introduces a numerical limitation on the time step needed for stability. Moreover, the whole domain needs to be updated each step, which is a time consuming operation for a sequential computer. The narrow band approach overcomes the last difficulty by limiting the computations to a narrow strip around the zero set. First suggested by Chopp [9], in the context of the level set method, and later developed in [1], the narrow band idea limits the computation to a tight strip of few grid points around the zero set. The rest of the domain serves only as a sign holder. As the curve evolves, the narrow band changes its shape and serves as a dynamic numerical support around the location of the zero level set.

B. The AOS scheme

Additive operator splitting (AOS) schemes were introduced by Weickert et al. [35] as an unconditionally stable numerical scheme for non-linear diffusion in image processing. Let us briefly review its main ingredients and adapt it to our model.

The original AOS model deals with the Perona-Malik [26], non-linear image evolution equation of the form $\partial_t u = \mathrm{div}\left(g(|\nabla u|)\nabla u\right)$ with given initial condition as the image $u(0) = u_0$. Let us re-write explicitly the right hand side of the evolution equation

$$\mathrm{div}\left(g(|\nabla u|)\nabla u\right) \quad = \quad \sum_{l=1}^{m} \partial_{x_l}\left(g(|\nabla u|)\partial_{x_l} u\right),$$

where l is an index running over the m dimensions of the problem, e.g., for a 2D image $m = 2$,$x_1 = x$, and $x_2 = y$.

As a first step towards discretization consider the operator

$$A_l(u^k) \quad = \quad \partial_{x_l}(g(|\nabla u^k|)\partial_{x_l}),$$

where the superscript k indicates the iteration number, e.g., $u^0 = u_0$. We can write the explicit scheme

$$u^{k+1} = \left[I + \tau \sum_{l=1}^{m} A_l(u^k) \right] u^k,$$

where, τ is the numerical time step. It requires an upper limit for τ if one desires to establish convergence to a stable steady state. Next, the semi-implicit scheme

$$u^{k+1} = \left[I - \tau \sum_{l=1}^{m} A_l(u^k) \right]^{-1} u^k,$$

is unconditionally stable, yet inverting the large bandwidth matrix is a computationally expensive operation.

Finally, the consistent, first order, semi-implicit, additive operator splitting scheme

$$u^{k+1} = \frac{1}{m}\sum_{l=1}^{m} \left[I - m\tau A_l(u^k) \right]^{-1} u^k,$$

may be applied to efficiently solve the non-linear diffusion.

The AOS semi-implicit scheme in $2D$ is then given by a linear tridiagonal system of equations

$$u^{k+1} = \frac{1}{2}\sum_{l=1}^{2}[I - 2\tau A_l(u^k)]^{-1}u^k, \qquad (1)$$

where $A_l(u^k)$ is a matrix corresponding to derivatives along the l-th coordinate axis. It can be efficiently solved for u^{k+1} by Thomas algorithm, see [35].

In our case, the geodesic active contour model is given by

$$\partial_t \phi = \mathrm{div}\left(g(|\nabla u_0|)\frac{\nabla \phi}{|\nabla \phi|} \right) |\nabla \phi|,$$

where u_0 is the image, and ϕ is the implicit representation of the curve. Since our interest is only at the zero level set of ϕ, we can reset ϕ to be a distance function every numerical iteration. One nice property of distance maps is it unit gradient magnitude almost everywhere. Thereby, the short term evolution for the geodesic active contour given by a distance map, with $|\nabla \phi| = 1$, is

$$\partial_t \phi = \mathrm{div}\left(g(|\nabla u_0|)\nabla \phi\right).$$

Note, that now $A_l(\phi^k) = A_l(u_0)$, which means that the matrices $[I - 2\tau A_l(u_0)]^{-1}$ can be computed once for the whole image. Yet, we need to keep the ϕ function as a distance map. This is done through re-initialization by Sethian's fast marching method every iteration.

It is now simple to introduce a weighted area 'balloon' like force to the scheme. The resulting AOS scheme with the 'balloon' then reads

$$\phi^{k+1} = \frac{1}{2}\sum_{l=1}^{2}[I - 2\tau A_l(u_0)]^{-1}(\phi^k + \tau\alpha g(u_0)), \qquad (2)$$

where α is the weighted area/balloon coefficient.

In order to reduce the computational cost we use a multi-scale approach [20]. We construct a Gaussian pyramid of the original image. The algorithm is first applied at the lower resolution. Next, the zero set is embedded at a higher resolution and the ϕ distance function is computed. Moreover, the computations are performed only within a limited narrow band around the zero set. The narrow band automatically modifies its shape as we re-initiate the distance map.

C. Re-initialization by the fast marching method

In order to maintain sub-grid accuracy, we detect the zero level set curve with sub-pixel accuracy. We apply a linear interpolation in the four pixel cells in which ϕ changes its sign. The grid points with the exact distance to the zero level set are then used to initialize the fast marching method.

Sethian's fast marching method [29], [28], is a computationally optimal numerical method for distance computation on rectangular grids. The method keeps a front of updated points sorted in a heap structure, and constructs a numerical solution iteratively, by fixing the smallest element at the top of the heap and expanding the solution to its neighboring grid points. This method enjoys a computational complexity bound of $O(N \log N)$, where N is the number of grid points in the narrow band. See also [8], [33], where consistent $O(N \log N)$ schemes are used to compute distance maps on rectangular grids.

III. Edge indicator functions for color and movies

What is a proper edge indicator for color images? Several generalizations for the gradient magnitude of gray level images were proposed, see e.g. [11], [27], [31]. In [25] Paragios and Deriche, introduced a probability based edge indicator function for movies. In this paper we have chosen the geometric philosophy to extract an edge indicator. We consider a measure suggested by the Beltrami framework in [31], to construct an edge indicator function.

A. Edges in Color

According to the Beltrami framework, a color image is considered as a two dimensional surface in the five dimensional spatial-spectral space. The metric tensor is used to measure distances on the image manifold. The magnitude of this tensor is an area element of the color image surface, which can be considered as a generalization of the gradient magnitude.

Formally, the metric tensor of the 2D image given by the 2D surface $\{x, y, R(x,y), G(x,y), B(x,y)\}$ in the $\{x, y, R, G, B\}$ space, is given by

$$(g_{ij}) = \begin{pmatrix} 1 + R_x^2 + G_x^2 + B_x^2 & R_xR_y + G_xG_y + B_xB_y \\ R_xR_y + G_xG_y + B_xB_y & 1 + R_y^2 + G_y^2 + B_y^2 \end{pmatrix},$$

where $R_x \equiv \partial_x R$. Our edge indicator is the largest eigenvalue of the structure tensor metric. It is the eigenvalue in the direction of maximal change in $dR^2 + dG^2 + dB^2$, and it reads

$$\lambda = 1 + \frac{1}{2}\sum_i |\nabla u^i|^2 + \sqrt{\left(\frac{1}{2}\sum_i |\nabla u^i|^2\right)^2 - \frac{1}{2}\sum_i\sum_j |\nabla u^i \times \nabla u^j|^2}$$

where $u^1 = R, u^2 = G, u^3 = B$. Then, the edge indicator function g is given by a decreasing function of λ, e.g., $g = (1+\lambda^2)^{-1}$.

B. Tracking objects in movies

Let us explore two possibilities to track objects in movies. The first, considers the whole movie volume as a Riemannian space, as done in [7]. In this case the active contour becomes an active surface. The AOS scheme in the spatial-temporal $3D$ hybrid space is

$$\phi^{k+1} = \frac{1}{3}\sum_l [I - 3\tau A_l(u_0)]^{-1}\phi^k,$$

where $A_l(u_0)$ is a matrix corresponding to derivatives along the l-th coordinate axis, where now $l \in [x, y, \mathcal{T}]$.

The edge indicator function is again derived from the Beltrami framework, where for color movies we pull-back the metric

$$(g_{ij}) = \begin{pmatrix} 1 + R_x^2 + G_x^2 + B_x^2 & R_xR_y + G_xG_y + B_xB_y & R_xR_\mathcal{T} + G_xG_\mathcal{T} + B_xB_\mathcal{T} \\ \cdot & 1 + R_y^2 + G_y^2 + B_y^2 & R_yR_\mathcal{T} + G_yG_\mathcal{T} + B_yB_\mathcal{T} \\ \cdot & \cdot & 1 + R_\mathcal{T}^2 + G_\mathcal{T}^2 + B_\mathcal{T}^2 \end{pmatrix}$$

which is the metric for a 3D volume in the 6D $\{x, y, \mathcal{T}, R, G, B\}$ spatial-temporal-spectral space. Now we have $\sqrt{\det(g_{ij})}dxdyd\mathcal{T}$ as a volume element of the image and, again, the largest eigenvalue of the structure tensor metric, λ, can be used as an edge indicator. Intuitively, the larger λ gets, the smaller spatial-temporal steps one should apply in order to cover the same volume.

A different approach uses the contour location in frame n as an initial condition for the 2D solution in frame $n+1$, see e.g. [3], [25]. The above edge indicator is still valid in this case. Note, that the aspect ratios between the time, the image space, and the intensity, should be determined according to the application.

The first approach was found to yield accurate results in off line tracking analysis. While the second approach gives up some accuracy, that is achieved by temporal smoothing in the first approach, for efficiency in real time tracking.

IV. Implementation details

There are some implementation considerations one should be aware of. For example, the summation over the two dimensions in the Equations (1,2) should be done in such a way that the matrices $A_1(u_0)$ and $A_2(u_0)$, corresponding to the derivatives along the x and y axes respectively, will be tridiagonal. This is achieved by spanning the matrix ϕ^k once by rows and once by columns.

Working on the whole domain is fairly straightforward. The matrix ϕ^k is represented as vectors ϕ_i^k corresponding to a single row/column. The vectors $[I - 2\tau A_l^i(u_0)]^{-1}\phi_i^k$, where A_l^i of size $|\phi_i^k| \times |\phi_i^k|$ is a matrix corresponding to the derivatives of a single row/column, are computed using the Thomas algorithm and summed according to their coordinates to yield the matrix ϕ^{k+1}.

Using the narrow band approach is a bit more tricky. The question is what is the most efficient representation for the narrow band that would allow to find for every row/column the segments that belong to the narrow band? One may suggest to use the run length encoding, but the standard static run length implementation is not sufficient. This is due to the fact that the narrow band is rebuilt every iteration using the fast marching algorithm, which generates narrow band pixels in arbitrary order. Creating

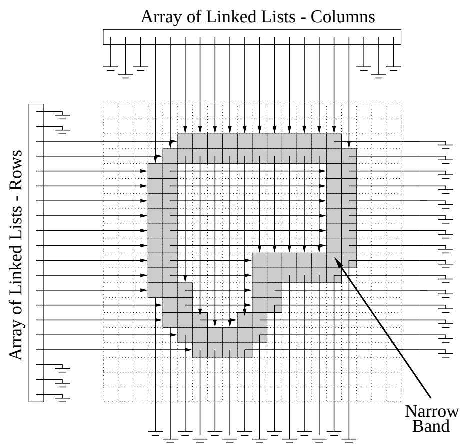

Fig. 1. Two arrays of linked lists used for narrow band representation.

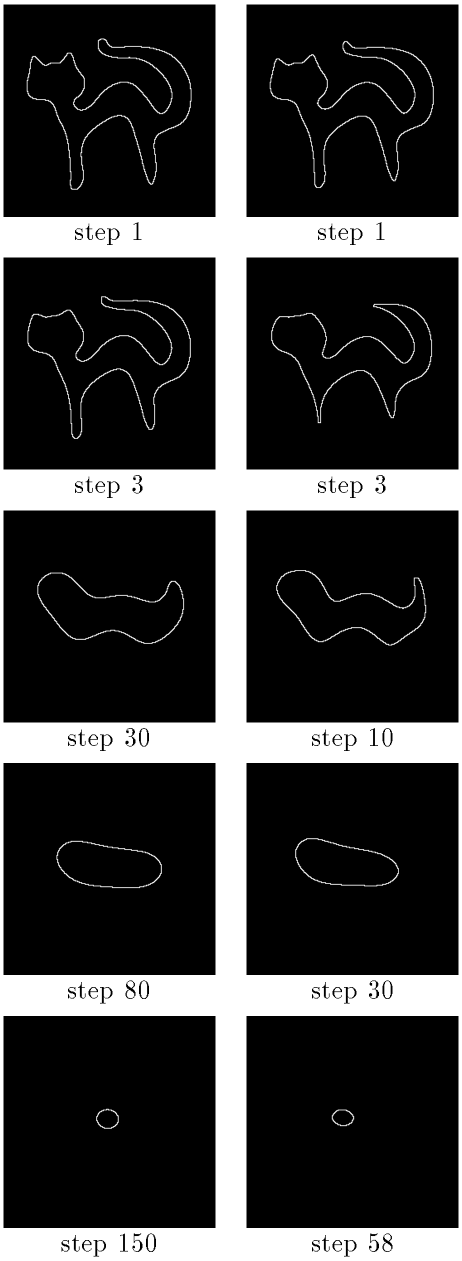

Fig. 2. Curvature flow by the proposed scheme. A non-convex curve vanishes in finite time at a circular point by Grayson's Theorem. The curve evolution is presented for two different time steps. Left: $\tau = 20$; Right: $\tau = 50$.

the run length encoding from such a stream of pixels can be done either off-line, by first constructing and then scanning a map of the whole image (which is clearly inefficient), or on-line using a dynamic data structure.

In our implementation we use two arrays of linked lists, one for the rows and one for the columns, where each linked list corresponds to one row/column and contains segments of adjacent pixels belonging to the narrow band (see Figure 1). Each segment is defined by its first and last pixels coordinates. Since a narrow band is generated as the fast marching algorithm grows outwards, adding a new pixel usually means just changing one of the boundaries of an

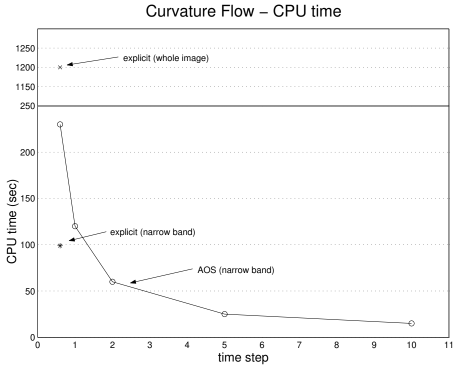

Fig. 3. Curvature flow CPU time for the explicit scheme and the implicit AOS scheme. First, the whole domain is updated, next, the narrow band is used to increase the efficiency, and finally the AOS speeds the whole process. For the explicit scheme the maximal time step that still maintains stability is choosen. For the AOS scheme, CPU times for several time steps are presented.

already existing segment. For reasonably simple contours the number of times a new segment is created or an existing one is merged with another is relatively low. Therefore, the number of segments per row/column is always small and the complexity of adding a new pixel to the narrow band is practically $O(1)$.

The calculations are performed separately for every horizontal and vertical segment ϕ_i^k and the vectors $[I - 2\tau A_l^i(u_0)]^{-1}\phi_i^k$, where A_l^i of size $|\phi_i^k| \times |\phi_i^k|$ is a matrix corresponding to the derivatives of a single segment, are summed according to their coordinates to yield the new level set function ϕ^{k+1} for the narrow band pixels only.

Another issue requiring a special attention is the value of the time step. If we choose a relatively large time step, the active contour may skip over the object boundary. The time step should thus correspond to the numerical support of the edges (edge width). This, in turn, dictates the width of the narrow band that should be wide enough, so that the contour would not escape it in one iteration. One way to overcome the time step limitation is to use a coarse to fine scales of boundary smoothing, with an appropriate time step for each scale. Finally, since the method is based on the AOS, which is a first-order approximation scheme, the numerical error grows linearly with the time step.

V. Experimental Results

As a simple example, the proposed method can be used as a consistent, unconditionally stable, and computationally efficient, numerical approximation for the curvature flow. The curvature flow, also known as curve shortening flow or geometric heat equation, is a well studied equation in the theory of curve evolution. It is proven to bring every simple closed curve into a circular point in finite time [15], [17]. Figure 2 shows an application of the proposed method for a curve evolving by its curvature and vanishes at a point. One can see how the number of iterations needed for the curve to converge to a point decreases as the time step is increased.

We tested several implementations for the curvature flow. Figure 3 shows the CPU time it takes the explicit and implicit schemes to evolve a contour into a circular point. For the explicit scheme we tested both the narrow band and the naive approach in which every grid point is updated every iteration. The tests were performed on an Ultra SPARC $360MHz$ machine for a 256×256 resolution image.

It should be noted that when the narrow band approach is used, the band width should be increased as the τ grows to ensure that the curve does not escape the band in one iteration.

Figure 4 shows multiple objects segmentation for a static color image. Here we used a balloon force to propagate the contour through the narrow passages between objects.

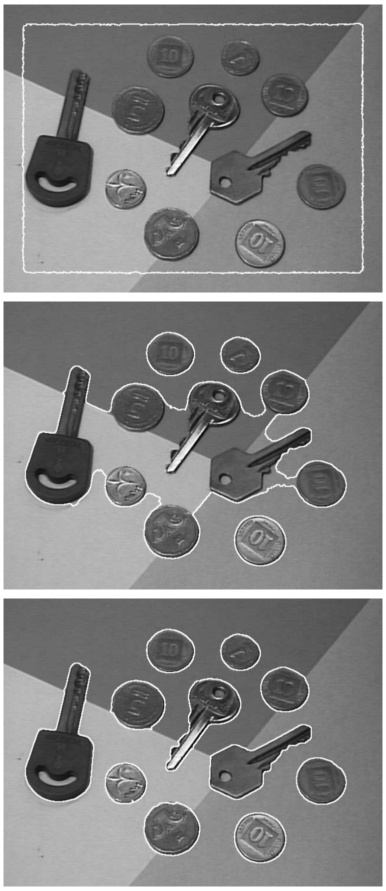

Fig. 4. Multiple objects segmentation in a static color image.

Figure 5 presents an example of medical image application, where the gray matter segmentation is performed on a single slice of a human head MRI image. The task is to detect a narrow layer of the brain bounded by two interfaces - the outer cortical surface (Cerebral Spinal Fluid(CSF)/gray matter interface) and the inner cortical surface (gray matter/white matter interface). The segmentation is performed by two active contours initialized inside the white matter regions. The negative balloon force coefficient is used to expand the contour towards the region boundary and the edge indicator functions are chosen to respond only to the edge points corresponding to the characteristic intensity profiles of the CSF/gray matter and the gray matter/white matter interfaces respectively. This specific medical problem introduces new challenging difficulties and possible solutions will be reported elsewhere [16].

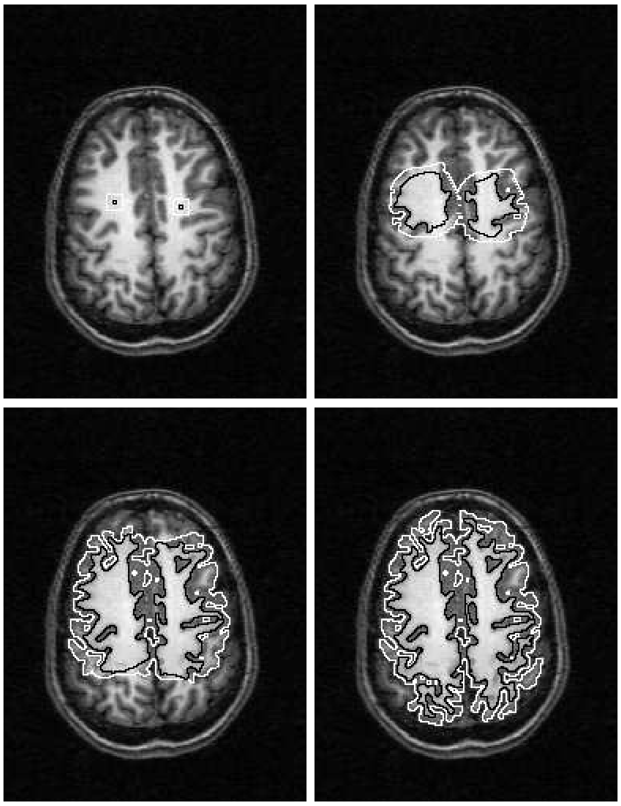

Fig. 5. Gray matter segmentation in a MRI brain image. The white contour converges to the outer cortical surface and the black contour converges to the inner cortical surface.

Figures 6 and 7 show segmentation results for color movies with difficult spatial textures. The tracking is performed at two resolutions. At the lower resolution we search for temporal edges and at the higher resolution we search for strong spatial edges. The contour found in the coarse grid is used as the initial contour at the fine grid.

It is possible to compute the inverse matrices of the AOS once for the whole image, or to invert small sub-matrices as new points enter or exit the narrow band. There is obviously a trade-off between the two approaches. For initialization, we have chosen the first approach, since the initial curve starts at the frame of the image and has to travel over most of the image until it captures the moving objects. While for tracking of moving objects in a movie, we use the local approach, since now the curve has only to adjust itself to local changes.

VI. Concluding Remarks

It was shown that an integration of advanced numerical techniques yield a computationally efficient algorithm that solves a geometric segmentation model. The numerical algorithm is consistent with the underlying continuous model. The proposed 'fast geodesic active contour' scheme

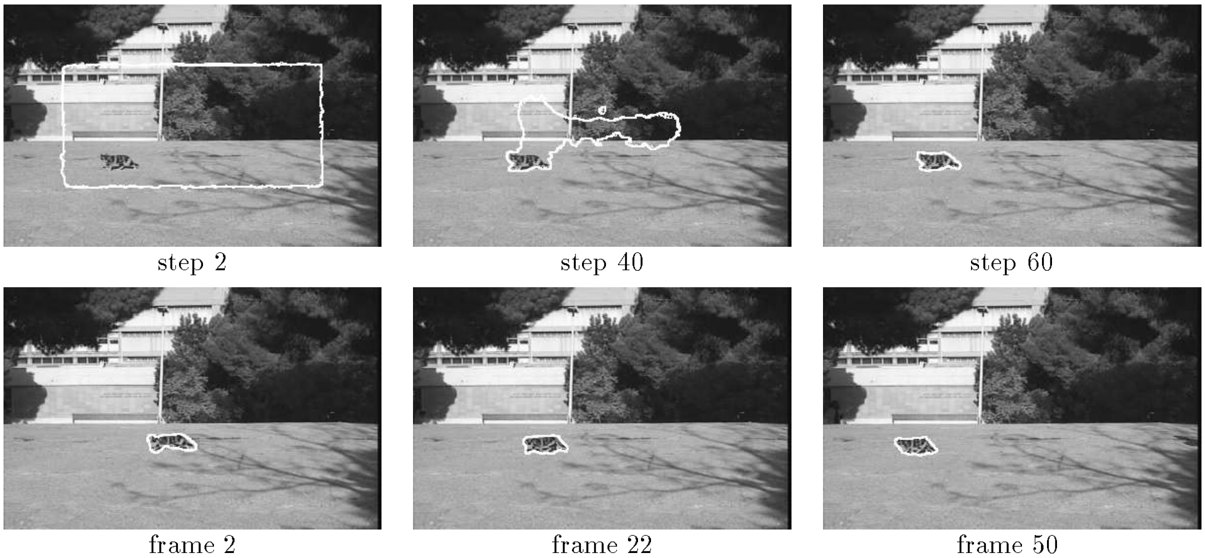

Fig. 6. Tracking a cat in a color movie by the proposed scheme. Top: Segmentation of the cat in a single frame. Bottom: Tracking the walking cat in the 50 frames sequence.

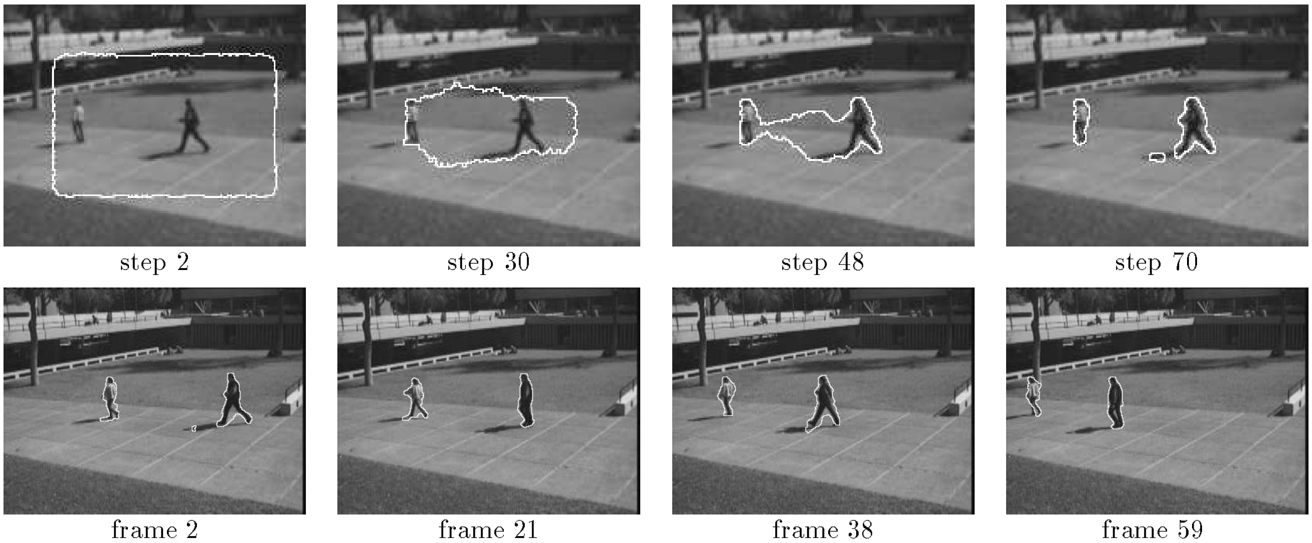

Fig. 7. Tracking two people in a color movie. Top: curve evolution in a single frame. Bottom: tracking two walking people in a 60 frame movie.

was applied successfully for image segmentation and tracking in movie sequences and color images. It combines the narrow band level set method, with adaptive operator splitting, and the fast marching.

VII. Acknowledgments

We thank Irad Yavneh for intriguing conversations and Nikolaos Paragios for useful correspondence.

References

[1] D Adalsteinsson and J A Sethian. A fast level set method for propagating interfaces. *J. of Comp. Phys.*, 118:269–277, 1995.

[2] V Caselles, F Catte, T Coll, and F Dibos. A geometric model for active contours. *Numerische Mathematik*, 66:1–31, 1993.

[3] V Caselles and B Coll. Snakes in movement. *SIAM J. Numer. Analy.*, 33(6):2445–2456, 1996.

[4] V Caselles, R Kimmel, and G Sapiro. Geodesic active contours. In *Proceedings ICCV'95*, pages 694–699, Boston, Massachusetts, June 1995.

[5] V Caselles, R Kimmel, and G Sapiro. Geodesic active contours. *IJCV*, 22(1):61–79, 1997.

[6] V Caselles, R Kimmel, G Sapiro, and C Sbert. Minimal surfaces: A geometric three dimensional segmentation approach. *Numerische Mathematik*, 77(4):423–425, 1997.

[7] V Caselles, R Kimmel, G Sapiro, and C Sbert. Minimal surfaces based object segmentation. *EEE Trans. on PAMI*, 19:394–398, 1997.

[8] C S Chiang, C M Hoffmann, and R E Lync. How to compute offsets without self-intersection. In *Proc. of SPIE*, volume 1620, page 76, 1992.

[9] D L Chopp. Computing minimal surfaces via level set curvature flow. *J. of Computational Physics*, 106(1):77–91, May 1993.

[10] L D Cohen. On active contour models and balloons. *CVGIP: Image Understanding*, 53(2):211–218, 1991.

[11] S Di Zenzo. A note on the gradient of a multi image. *Computer Vision, Graphics, and Image Processing*, 33:116–125, 1986.

[12] B A Dubrovin, A T Fomenko, and S P Novikov. *Modern Geom-*

etry - Methods and Applications I. Springer-Verlag, New York, 1984.

[13] O Faugeras and R Keriven. Variational principles, surface evolution PFE's, level set methods, and the stereo problem. *IEEE Trans. on Image Processing*, 7(3):336–344, 1998.

[14] P Fua and Y G Leclerc. Model driven edge detection. *Machine Vision and Applications*, 3:45–56, 1990.

[15] M Gage and R S Hamilton. The heat equation shrinking convex plane curves. *J. Diff. Geom.*, 23, 1986.

[16] R Goldenberg, R Kimmel, E Rivlin, and M Rudzsky. Cortex segmentation - a fast variational geometric approach. In *Proc. of IEEE workshop on Variational and Level Set Methods in Computer Vision.*, July 2001. To appear.

[17] M A Grayson. The heat equation shrinks embedded plane curves to round points. *J. Diff. Geom.*, 26, 1987.

[18] M Kass, A Witkin, and D Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331, 1988.

[19] S Kichenassamy, A Kumar, P Olver, A Tannenbaum, and A Yezzi. Gradient flows and geometric active contour models. In *Proceedings ICCV'95*, Boston, Massachusetts, June 1995.

[20] B Leroy, I L Herlin, and L Cohen. Multi-resolution algorithms for active contour models. In *Proc. 12th Int. Conf. on Analysis and Optimization of Systems (ICAOS'96)*, Paris, France, June 1996.

[21] R Malladi, R Kimmel, D Adalsteinsson, V Caselles, G Sapiro, and J A Sethian. A geometric approach to segmentation and analysis of 3D medical images. In *Proceedings of IEEE/SIAM workshop on Biomedical Image Analysis*, San-Francisco, California, June 1996.

[22] R Malladi and J A Sethian. An O(N log N) algorithm for shape modeling. *Proceedings of National Academy of Sciences, USA*, 93:9389–9392, 1996.

[23] R Malladi, J A Sethian, and B C Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Trans. on PAMI*, 17:158–175, 1995.

[24] S J Osher and J A Sethian. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. of Comp. Phys.*, 79:12–49, 1988.

[25] N Paragios and R Deriche. A PDE-based level set approach fro detection and tracking of moving objects. In *Proc. of the 6th ICCV*, Bombay, India, 1998.

[26] P Perona and J Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE-PAMI*, 12:629–639, 1990.

[27] G Sapiro and D L Ringach. Anisotropic diffusion of multivalued images with applications to color filtering. *IEEE Trans. Image Proc.*, 5:1582–1586, 1996.

[28] J A Sethian. *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences.* Cambridge Univ. Press, 1996.

[29] J A Sethian. A marching level set method for monotonically advancing fronts. *Proc. Nat. Acad. Sci.*, 93(4), 1996.

[30] J Shah. A common framework for curve evolution, segmentation and anisotropic diffusion. In *Proceedings IEEE CVPR'96*, pages 136–142, 1996.

[31] N Sochen, R Kimmel, and R Malladi. A general framework for low level vision. *IEEE Trans. on Image Processing*, 7(3):310–318, 1998.

[32] D Terzopoulos, A Witkin, and M Kass. Constraints on deformable models: Recovering 3D shape and nonrigid motions. *Artificial Intelligence*, 36:91–123, 1988.

[33] J N Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Trans. on Automatic Control*, 40(9):1528–1538, 1995.

[34] J Weickert. Fast segmentation methods based on partial differential equations and watershed transformation. In *Mustererkennung*, pages 93–100. Springer, Berlin, 1998.

[35] J Weickert, B M ter Haar Romeny, and M A Viergever. Efficient and reliable scheme for nonlinear diffusion filtering. *IEEE Trans. on Image Processing*, 7(3):398–410, 1998.

[36] S C Zhu. *Statistical and Copmutational Theories for Image Segmentation, Texture Modeling and Object Recognition.* Ph.D. thesis, Harvard University, January 1996.

[37] S C Zhu, A Yuille, and T S Lee. Region competition: Unifying snakes, region growing, and bayes/mdl for multi-band image segmentation. In *ICCV95*, pages 416–423, 1995.