Fast Edge Integration

Ron Kimmel

ABSTRACT A two–dimensional variational explanation for the Marr-Hildreth and the Haralick-Canny like edge detectors was recently presented in [16, 17]. For example, the zero crossings of the image Laplacian were shown to be optimal edge integration curves that solve a geometric problem. These are the curves whose normals best align with the image gradient field. Based on these observations, an improved active contour model was suggested, and its performances were shown to be better than classical geodesic active contours when directional information about the edge location is provided. We present a general model that incorporates the alignment as part of other driving forces of an active contour, together with the geodesic active contour model for regularization, and the minimal variance criterion suggested by Chan and Vese [5]. The minimal variance functional is related to segmentation by a threshold and to the Max-Lloyd quantization procedure. Here, we integrate all these geometric measures as part of one unified framework. Finally, we introduce unconditionally stable and simple numerical schemes for efficiently implementing the improved geometric active contour edge integration procedure.

1 Introduction

We start with a brief history of edge detection and relate it to modern geometry and variational principles. The most simple edge detectors try to locate points defined by local maxima of the image gradient magnitude. The Marr and Hildreth edges are a bit more sophisticated, and were defined as the zero crossing curves of a Laplacian of Gaussian (LoG) applied to the image [22, 21]. The Marr-Hildreth edge detection and integration process can be regarded as a way to determine curves in the image plane that pass through points where the gradient is high and whose normal direction best aligns with the local edge direction as predicted by the image gradient. This observation was first presented in [16]. The importance of orientation information in a variational setting for delicate segmentation tasks was recently also thought of by Vasilevskiy and Siddiqi [28]. They used alignment with a general vector field as a segmentation criterion of complicated closed thin structures in 3D medical images. In [17] it was shown that the Haralick edge detector [14, 2], which is the main procedure in the Canny edge detector, can be interpreted as a solution of a two–dimensional variational principle that combines the alignment term with

a topological homogeneity measure. We will not explore this observation here, as we have found the geodesic active contour to have somewhat better regularization performances in most cases of geometric active contour edge integration processes, which is the main topic of this chapter.

Section 2 introduces some of the mathematical notations we use in this chapter. In Section 3, we formulate the idea of geometric curve evolution for segmentation, and review various types of variational measures (geometric functionals). These functionals describe an integral quantity defined by the curve. Our goal would be to search for a curve that maximizes or minimizes these integral measures. Next, in Section 4 we compute the first variation of each of these functionals, and comment on how to use it in a dynamic gradient descent curve evolution process. Section 5 gives the level set formulation for the various curve evolution procedures. In Section 6, motivated by [12], we present an efficient numerical scheme that couples an alternating direction implicit multiplicative scheme, with narrow band [8, 1], and re-distancing via the fast marching method [26]. The scheme is unconditionally stable and thus allows large time steps for fast convergence.

2 Mathematical Notations

Consider a gray level image as a function $I : \Omega \to I\!R^+$ where $\Omega \in I\!R^2$ is the image domain. The image gradient vector field is given by $\nabla I(x,y) \equiv \{I_x, I_y\}$, were we used subscripts to denote the partial derivatives in this case, e.g., $I_x \equiv \partial I(x,y)/\partial x$. We search for a contour, $C : [0, L] \to I\!R^2$, given in a parametric form $C(s) = \{x(s), y(s)\}$, where s is an arclength parameter, and whose normal is defined by $\vec{n}(s) = \{-y_s(s), x_s(s)\}$. This contour somehow interacts with the given image, for example, a curve whose normal aligns with the gradient vector field, where the alignment of the two vectors can be measured by their inner product that we denote by $\langle \vec{n}, \nabla I \rangle$. We also use subscripts to denote full derivatives, such that the curve tangent is given by $C_s = \{x_s, y_s\} = \{dx(s)/ds, dy(s)/ds\}$. In some cases we will also use p to indicate an arbitrary (non-geometric) parameterization of the planar curve. In which case, the tangent is $C_s = C_p/|C_p|$, and the normal can be written as

$$\vec{n} = \frac{\{-y_p, x_p\}}{|C_p|},$$

where $|C_p| = \sqrt{x_p^2 + y_p^2}$. We have the relation between the arclength s and a general arbitrary parameter p, given by

$$ds = \sqrt{dx^2 + dy^2} = \sqrt{\left(\frac{dx(p)}{dp}\right)^2 + \left(\frac{dy(p)}{dp}\right)^2}\, dp = |C_p| dp.$$

Define, as usual, κ to be the curvature of the curve C, and the curvature

vector $\kappa\vec{n} = C_{ss}$. We also define Ω_C to be the domain inside the curve C, see Figure 1.

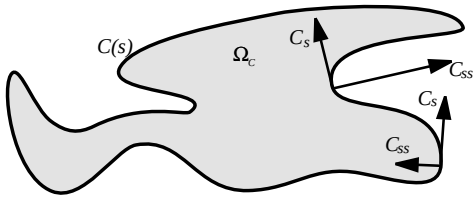

FIGURE 1. A closed curve C, with C_s the unit tangent, $\kappa\vec{n} = C_{ss}$ the curvature vector, and Ω_C the area inside the curve.

In this chapter we deal with two types of integral measures that are related via the Green theorem. The first is defined along the curve by the general form of

$$E(C) = \int_0^L g(C(s))ds.$$

The second functional integrates the values of the function $f(x, y)$ inside the curve, and is usually referred to as a region based measure,

$$E(C) = \iint_{\Omega_C} f(x, y)dxdy,$$

where Ω_C is the region inside the curve C. Formally, we search for the optimal planar curve C, such that

$$C = \arg\max_C E(C).$$

3 Geometric Integral Measures for Active Contours

The evolution of dynamic edge integration processes and active contours started with the classical snakes [15], followed by non-variational geometric active contours [20, 3], and more recent geodesic active contours [4]. Here, we restrict our discussion to parameterization invariant (geometric) functionals, that do dot depend on the internal parameterization of the curve, but rather on its geometry and the properties of the image. From these functionals we extract the first variation, and use it as a gradient descent process, also known as geometric active contour.

3.1 Alignment Term

Consider the geometric functional

$$E_A(C) = \int_0^L \langle \nabla I(x(s), y(s)), \vec{n}(s) \rangle ds,$$

or in its 'robust' form

$$E_{AR}(C) = \int_0^L |\langle \nabla I(x(s), y(s)), \vec{n}(s) \rangle| ds,$$

where the absolute value of the inner product between the image gradient and the curve normal is our alignment measure. See Figure 2. The motiva-

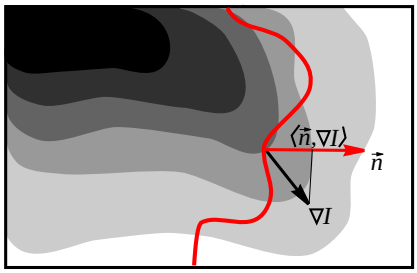

FIGURE 2. The curve C, its normal $\vec{n}$ at a specific point, and the image gradient ∇I at that point. The alignment term integrates the projection of ∇I on the normal along the curve.

tion is the fact that in many cases, the gradient direction is a good estimator for the orientation of the edge contour. The inner product gets high values if the curve normal aligns with the image gradient direction. This measure also uses the gradient magnitude as an edge indicator. Therefore, our goal would be to find curves that maximize this geometric functional.

3.2 Weighted Region

In some cases we have a quantity we would like to maximize by integration inside the region Ω_C, defined by the curve C. In its most general setting, this weighted area measure is

$$E_W(C) = \iint_{\Omega_C} f(x, y) dx dy,$$

where $f(x, y)$ is any scalar function. A simple example is $f(x, y) = 1$, for which the functional $E(C)$ measures the area inside the curve C, that is, the area of the region Ω_C that we also denote by $|\Omega_C|$.

3.3 Minimal Variance

In [5], Chan and Vese proposed a minimal variance criterion, given by

$$E_{MV}(C, c_1, c_2) = \frac{1}{2}\iint_{\Omega_C}(I(x,y)-c_1)^2 dxdy + \frac{1}{2}\iint_{\Omega\setminus\Omega_C}(I(x,y)-c_2)^2 dxdy.$$

As we will see, in the optimal case, the two constants, c_1 and c_2, get the mean intensities in the interior (inside) and the exterior (outside) the contour C, respectively. The optimal curve would best separate the interior and exterior with respect to their relative average values. In the optimization process we look for the best separating curve, as well as for the optimal expected values c_1 and c_2. Such optimization problems, in higher dimensions, are often encountered in color quantization and classification problems.

In order to control the smoothness of their active contour, Chan and Vese also included the arclength $\int ds$ as a regularization term. Here we propose to use the more general weighted arclength, $\int g(C(s))ds$, also known as the geodesic active contour functional [4], as a data sensitive regularization term. It can be shown to yield better results in most cases and simplifies to the regularization used by Chan and Vese for the selection of $g = 1$.

One could consider more generic region based measures like

$$E(C) = \iint_{\Omega_C}\|T(I(x,y)) - \bar{c_1}\|_{L_p} dxdy + \iint_{\Omega\setminus\Omega_C}\|T(I(x,y)) - \bar{c_2}\|_{L_p} dxdy$$

where T is a general transformation of the image, the norm $\|\cdot\|_{L_p}$ can be chosen according to the problem in hand, and $\bar{c_1}$ and $\bar{c_2}$ are vectors of possible parameters. One such example is the robust measure

$$E_{RMV}(C) = \iint_{\Omega_C}|I(x,y) - c_1| dxdy + \iint_{\Omega\setminus\Omega_C}|I(x,y) - c_2| dxdy.$$

3.4 Geodesic Active Contour

The geodesic active contour [4] model is defined by the functional

$$E_{GAC}(C) = \int_0^L g(C(s))ds.$$

It is an integration of an inverse edge indicator function, like $g(x,y) = 1/(1+|\nabla I|^2)$, along the contour. The search, in this case, would be for a curve along which the inverse edge indicator gets the smallest possible values. That is, we would like to find the curve C that minimizes this functional. The geodesic active contour was shown in [16, 17] to serve as a good regularization for other dominant terms like the minimal variance in noisy images, or the alignment term in cases we have good orientation estimation of the edge. A well studied example is $g(x,y) = 1$, for which the functional measures the total arclength of the curve.

4 Calculus of Variations for Geometric Measures

Given a curve integral of the general form,

$$E(C) = \oint_C L(C_p, C)dp,$$

where p is an arbitrary parameter, we compute the first variation by

$$\frac{\delta E(C)}{\delta C} = \begin{pmatrix} \frac{\partial}{\partial x} - \frac{d}{dp}\frac{\partial}{\partial x_p} \\ \frac{\partial}{\partial y} - \frac{d}{dp}\frac{\partial}{\partial y_p} \end{pmatrix} L(C, C_p).$$

The extremals of the functional $E(C)$ can be identified by the Euler Lagrange equation $\delta E(C)/\delta C = 0$. A dynamic process known as gradient descent, that takes an arbitrary curve towards a maximum of $E(C)$, is given by the curve evolution equation

$$\frac{\partial C}{\partial t} = \frac{\delta E(C)}{\delta C},$$

where we added a virtual 'time' parameter t to our curve to allow its evolution into a family of planar curves $C(s,t)$. Our hope is that this evolution process would take an almost arbitrary initial curve into a desired configuration, which gives a significant extremum of our functional. In this chapter, we restrict ourselves to closed contours. When considering open contours, one should also handle the end points and add additional constraints to determine their optimal locations, as done for example in [11, 16].

Lemma 1 *Given the vector field $\vec{V}(x,y) = \{u(x,y), v(x,y)\}$, we have the alignment measure,*

$$E_A(C) = \oint_C \langle \vec{V}, \vec{n} \rangle ds$$

for which the first variation is given by

$$\frac{\delta E(C)}{\delta C} = \mathrm{div}(\vec{V})\vec{n}.$$

Proof. We first change the integration variable from an arclength s to an arbitrary parameter p.

$$\begin{aligned} E_A(C) = \int_0^L \langle \vec{V}, \vec{n} \rangle ds &= \int_0^L \langle \{u, v\}, \{-y_s, x_s\} \rangle ds \\ &= \int_0^1 \left\langle \{u, v\}, \frac{\{-y_p, x_p\}}{|C_p|} \right\rangle |C_p| dp \\ &= \int_0^1 (v x_p - u y_p) dp. \end{aligned}$$

Next, we compute the first variation for the x component,

$$\begin{aligned}\frac{\delta E_A(C)}{\delta x} = \left(\frac{\partial}{\partial x} - \frac{d}{dp}\frac{\partial}{\partial x_p}\right)(vx_p - uy_p) &= v_x x_p - u_x y_p - \frac{d}{dp}v \\ &= v_x x_p - u_x y_p - v_x x_p - v_y y_p \\ &= -y_p(u_x + v_y) = -y_p \mathrm{div}(\vec{V}).\end{aligned}$$

Similarly, for the y component we have $\frac{\delta E_A}{\delta y} = x_p \mathrm{div}(\vec{V})$. By freedom of parameterization, we end up with the first variation, $\frac{\delta E_A}{\delta C} = \mathrm{div}(\vec{V})\vec{n}$. ■

An important example is $\vec{V} = \nabla I$, for which we have as first variation

$$\frac{\delta E_A(C)}{\delta C} = \Delta I \vec{n},$$

where $\Delta I = I_{xx} + I_{yy}$ is the image Laplacian. The Euler Lagrange equation $\delta E_A / \delta C = 0$ gives a variational explanation for Marr-Hildreth edge detector that is defined by the zero crossings of the Laplacian, as first reported in [16].

Lemma 2 *The robust alignment term given by*

$$E_{AR}(C) = \oint_C |\langle \vec{V}, \vec{n} \rangle| ds,$$

yields the first variation

$$\frac{\delta E_{AR}(C)}{\delta C} = \mathrm{sign}(\langle \vec{V}, \vec{n}(s) \rangle) \mathrm{div}(\vec{V}) \vec{n},$$

Proof. We start by changing to an arbitrary parameter,

$$\begin{aligned}E_{AR}(C) = \int_0^L |\langle \vec{V}, \vec{n} \rangle| ds &= \int_0^1 \left| \left\langle \{v, u\}, \frac{\{-y_p, x_p\}}{|C_p|} \right\rangle \right| |C_p| dp \\ &= \int_0^1 \sqrt{(ux_p - vy_p)^2} dp.\end{aligned}$$

Next, we compute the first variation for the x component,

$$\begin{aligned}\frac{\delta E_{AR}(C)}{\delta x} &= \left(\frac{\partial}{\partial x} - \frac{d}{dp}\frac{\partial}{\partial x_p}\right)\sqrt{(vx_p - uy_p)^2} \\ &= -y_p \mathrm{sign}(vx_p - uy_p)(u_x + v_y) = -y_p \mathrm{sign}(\langle \vec{V}, \vec{n} \rangle) \mathrm{div}(\vec{V}).\end{aligned}$$

Similarly, for the y component we have $\frac{\delta E_{AR}}{\delta y} = x_p \mathrm{sign}(\langle \vec{V}, \vec{n} \rangle) \mathrm{div}(\vec{V})$. By freedom of parameterization, we end up with the first variation, $\frac{\delta E_{AR}}{\delta C} = \mathrm{sign}(\langle \vec{V}, \vec{n} \rangle) \mathrm{div}(\vec{V}) \vec{n}$. ■

An important example is $\vec{V} = \nabla I$, for which we have

$$\frac{\delta E_{AR}(C)}{\delta C} = \mathrm{sign}(\langle \nabla I, \vec{n}(s) \rangle) \Delta I \vec{n}.$$

Lemma 3 *The weighted region functional*

$$E_W(C) = \iint_{\Omega_C} f(x,y)dxdy,$$

yields the first variation

$$\frac{\delta E_W(C)}{\delta C} = -f(x,y)\vec{n}.$$

Proof. Following [30], we define the two functions $P(x,y)$ and $Q(x,y)$, such that $P_y(x,y) = -\frac{1}{2}f(x,y)$ and $Q_x = \frac{1}{2}f(x,y)$. We readily have that $f(x,y) = Q_x - P_y$. Next, using Green's theorem we can write

$$\begin{aligned} E(C) = \iint_{\Omega_C} f(x,y)dxdy &= \iint_{\Omega_C} (Q_x - P_y)dxdy \\ &= \int_C (Pdx + Qdy) \\ &= \int_C (Px_s + Qy_s)ds = \int_C \langle \{-Q,P\}, \vec{n}\rangle ds, \end{aligned}$$

for which the first variation is given by Lemma 1, for $\vec{V} = \{-Q,P\}$, as

$$\frac{\delta E(C)}{\delta C} = \mathrm{div}(\{-Q,P\})\vec{n} = -(Q_x - P_y)\vec{n} = -f\vec{n}.$$

■

This term is sometimes called the weighted area [30] term, and for f constant, its resulting variation is known as the 'balloon' [9] force. If we set $f = 1$, the gradient descent curve evolution process is the constant flow. It generates offset curves via $C_t = \vec{n}$, and its efficient implementation is closely related to Euclidean distance maps [10, 6] and fast marching methods [26].

Lemma 4 *The geodesic active contour model is*

$$E_{GAC}(C) = \oint_C g(C(s))ds,$$

for which the first variation is given by

$$\frac{\delta E_{GAC}(C)}{\delta C} = -(\kappa g - \langle \nabla g, \vec{n}\rangle)\vec{n}.$$

Proof.

$$\begin{aligned} E_{GAC}(C) = \int_0^L g(C(s))ds &= \int_0^1 g(C(p))|C_p|dp \\ &= \int_0^1 g(C(p))\sqrt{x_p^2 + y_p^2}dp. \end{aligned}$$

Next, we compute the first variation for the x component,

$$\begin{aligned}\frac{\delta E_{GAC}(C)}{\delta x} &= \left(\frac{\partial}{\partial x} - \frac{d}{dp}\frac{\partial}{\partial x_p}\right)\left(g(x(p),y(p))\sqrt{x_p^2+y_p^2}\right)\\ &= g_x|C_p| - \frac{d}{dp}g\frac{x_p}{\sqrt{x_p^2+y_p^2}}\\ &= g_x|C_p| - (g_x x_p + g_y y_p)\frac{x_p}{|C_p|}\\ &\quad -g\frac{x_{pp}|C_p| - x_p(x_p x_{pp} + y_p y_{pp})/|C_p|}{|C_p|^2}\\ &= y_p(\kappa g - \langle\nabla g, \vec{n}\rangle).\end{aligned}$$

where we used the curvature defined by

$$\kappa = \frac{x_{pp}y_p - y_{pp}x_p}{|C_p|^3}.$$

Similarly, for the y component we have $\frac{\delta E_{GAC}}{\delta y} = -x_p(\kappa g - \langle\nabla g, \vec{n}\rangle)$. By freedom of parameterization we end up with the above first variation. ■

We will use this term mainly for regularization. If we set $g = 1$, the gradient descent curve evolution result given by $C_t = -\delta E_{GAC}(C)/\delta C$ is the well known curvature flow, $C_t = \kappa\vec{n}$ or equivalently $C_t = C_{ss}$, also known as the geometric heat equation.

Lemma 5 *The Chan-Vese minimal variance criterion [5] is given by*

$$E_{MV}(C, c_1, c_2) = \frac{1}{2}\iint_{\Omega_C}(I-c_1)^2 dxdy + \frac{1}{2}\iint_{\Omega\setminus\Omega_C}(I-c_2)^2 dxdy,$$

for which the first variation is

$$\begin{aligned}\frac{\delta E_{MV}}{\delta C} &= (c_2-c_1)\left(I - \frac{c_1+c_2}{2}\right)\vec{n}\\ \frac{\delta E_{MV}}{\delta c_1} &= \iint_{\Omega_C} I dxdy - c_1\iint_{\Omega_C} dxdy\\ \frac{\delta E_{MV}}{\delta c_2} &= \iint_{\Omega\setminus\Omega_C} I dxdy - c_2\iint_{\Omega\setminus\Omega_C} dxdy.\end{aligned}$$

Proof. Using Lemma 3, we have the first variation given by

$$\begin{aligned}\frac{\delta E_{MV}}{\delta C} &= \frac{1}{2}\left((I-c_1)^2 - (I-c_2)^2\right)\vec{n}\\ &= \frac{1}{2}\left(I^2 - 2Ic_1 + c_1^2 - I^2 + 2Ic_2 - c_2^2\right)\vec{n}\\ &= \left((c_2-c_1)I - \frac{(c_1+c_2)(c_2-c_1)}{2}\right)\vec{n}\end{aligned}$$

$$= \quad (c_2 - c_1)\left(I - \frac{c_1 + c_2}{2}\right)\vec{n}.$$

The optimal c_1 and c_2, extracted from $\delta E_{MV}/\delta c_1 = 0$ and $\delta E_{MV}/\delta c_2 = 0$, are the average intensities of the image inside and outside the contour, respectively. ■

One could recognize the variational interpretation of segmentation by the threshold $(c_1 + c_2)/2$ given by the Euler Lagrange equation $\delta E_{MV}/\delta C = 0$.

Finally, we treat the robust minimal deviation measure E_{RMV}.

Lemma 6 *The robust minimal total deviation criterion is given by*

$$E_{RMV}(C, c_1, c_2) = \iint_{\Omega_C} |I - c_1| dxdy + \iint_{\Omega \setminus \Omega_C} |I - c_2| dxdy,$$

for which the first variation is

$$\begin{aligned}
\frac{\delta E_{RMV}}{\delta C} &= (|I - c_1| - |I - c_2|)\,\vec{n} \\
\frac{\delta E_{RMV}}{\delta c_1} &= \iint_{\Omega_C} \text{sign}(I - c_1) dxdy \\
\frac{\delta E_{RMV}}{\delta c_2} &= \iint_{\Omega \setminus \Omega_C} \text{sign}(I - c_2) dxdy
\end{aligned}$$

Proof. Using Lemma 3, we have the first variation $\frac{\delta E_{RMV}(C)}{\delta C}$.

The optimal c_1 and c_2, extracted from $\delta E_{RMV}/\delta c_1 = 0$ and $\delta E_{RMV}/\delta c_2 = 0$, are the median intensities of the image inside and outside the contour, respectively:

$$\begin{aligned}
\frac{\delta E_{RMV}}{\delta c_1} &= \frac{d}{dc_1} \iint_{\Omega_C} \sqrt{(I - c_1)^2} dxdy \\
&= -\iint_{\Omega_C} \frac{I - c_1}{|I - c_1|} dxdy \\
&= -\iint_{\Omega_C} \text{sign}(I - c_1) dxdy.
\end{aligned}$$

We have that $\iint_{\Omega_C} \text{sign}(I - c_1) dxdy = 0$ for the selection of c_1 as the value of $I(x, y)$ in Ω_C that splits its area into two equal parts. From obvious reasons we refer to this value as the median of I in Ω_C, or formally,

$$\begin{aligned}
c_1 &= \text{median}_{\Omega_C} I(x, y) \\
c_2 &= \text{median}_{\Omega \setminus \Omega_C} I(x, y).
\end{aligned}$$

■

The computation of c_1 and c_2 can be efficiently implemented via the intensity histograms in the interior or the exterior of the contour. One

rough discrete approximation is the median of the pixels inside or outside the contour.

The robust minimal deviation term should be preferred when the penalty for isolated pixels with wrong affiliation is insignificant. The minimal variance measure penalizes large deviations in a quadratic fashion and would tend to over-segment such events or require large regularization that could over-smooth the desired boundaries.

5 Gradient Descent in Level Set Formulation

We embed a closed curve in a higher dimensional $\phi(x, y)$ function, which implicitly represents the curve C as a zero set, i.e., $C = \{\{x, y\} : \phi(x, y) = 0\}$. This way, the well known Osher-Sethian [23] level-set method can be employed to implement the curve propagation toward its optimal location. Figure 3 presents a planar curve, and two implicit representations for this curve, displayed in two different ways.

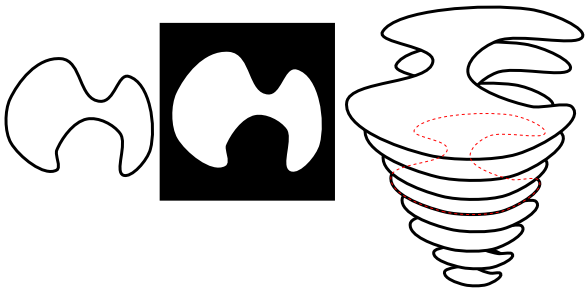

FIGURE 3. Left: A given planar curve. Middle: An implicit representation of the curve as a threshold set of a gray level image. Right: An implicit representation of the curve as a level set of a smooth function displayed by its level set contours.

Given the curve evolution equation $C_t = \gamma\vec{n}$, its implicit level set evolution equation reads

$$\phi_t = \gamma|\nabla\phi|.$$

The equivalence of these two evolutions can be easily verified using the chain rule and the relation $\vec{n} = \nabla\phi/|\nabla\phi|$,

$$\phi_t \quad = \quad \phi_x x_t + \phi_y y_t = \langle\nabla\phi, C_t\rangle = \langle\nabla\phi, \gamma\vec{n}\rangle = \gamma\left\langle\nabla\phi, \frac{\nabla\phi}{|\nabla\phi|}\right\rangle = \gamma|\nabla\phi|.$$

Those familiar with the optical flow problem in image analysis could easily recognize this derivation. There is an interesting relation between the

classical optical flow problem and the level set method. Level set formulation for the evolution of a family of embedded curves can be interpreted as a dynamic image synthesis process. On the other hand, optical flow in image analysis is a search for the motion of features in the image. These two inverse problems share the same fundamental equation. Computing a vector field that represents the flow of the gray level sets from a given sequence of images is known as the 'normal flow' computation. Next, at a higher level of image understanding, the motion field of objects in an image is known as the 'optical flow'. On the other hand in the level set formulation, the goal is to compute the dynamics of an arbitrary image, in which one level set represents a specific curve, from a given motion vector field of that specific level set. The image in this case is an implicit representation of its level sets, while the vector field itself could be extracted from either the geometric properties of the level sets, or from another image or external data.

As a first example, consider the explicit curve evolution as a gradient descent flow for the robust alignment term, given by

$$C_t = \text{sign}(\langle \nabla I, \vec{n} \rangle) \Delta I \vec{n}.$$

The corresponding level set evolution is

$$\phi_t = \text{sign}(\langle \nabla I, \nabla \phi \rangle) \Delta I |\nabla \phi|.$$

We can now add to our model the geodesic active contour term, the threshold term, or its dynamic expectation version defined by the minimal variance criterion. The optimization problem takes the form of

$$\arg\max_{C,c_1,c_2} E(C, c_1, c_2),$$

for the functional

$$\begin{aligned} E(C, c_1, c_2) &= E_{AR}(C, c_1, c_2) - \alpha E_{GAC}(C) - \beta E_{MV}(C) \\ &= \oint_C |\langle \nabla I, \vec{n} \rangle| ds - \alpha \oint_C g(C(s)) ds \\ &\quad -\beta \frac{1}{2} \left(\iint_{\Omega_C} (I - c_1)^2 dx dy + \iint_{\Omega \setminus \Omega_C} (I - c_2)^2 dx dy \right), \end{aligned}$$

where α and β are constants, and α is small so that the geodesic active contour term is used mainly for regularization. The first variation as a gradient descent process, is

$$\begin{aligned} C_t &= [\text{sign}(\langle \vec{n}, \nabla I \rangle) \Delta I + \alpha \left(g(x,y)\kappa - \langle \nabla g, \vec{n} \rangle \right) \\ &\qquad + \beta (c_2 - c_1) \left(I - (c_1 + c_2)/2 \right)] \vec{n}. \\ c_1 &= \frac{1}{|\Omega_C|} \iint_{\Omega_C} I(x,y) dx dy \end{aligned}$$

$$c_2 = \frac{1}{|\Omega \setminus \Omega_C|} \iint_{\Omega \setminus \Omega_C} I(x,y)dxdy,$$

where $|\Omega_C|$ denotes the area of the regions Ω_C. One could recognize the relation to Max-Lloyd quantization method, as the simplest implementation for this system is a sequential process that involves a change of the segmentation set followed by an update of the mean representing each set. The difference is the additional penalties and resulting forces we use for the smoothness and alignment of the boundary contours.

The level set formulation of the curve evolution equation is

$$\phi_t = \left[\text{sign}(\langle \nabla\phi, \nabla I \rangle)\Delta I + \alpha \text{div}\left(g(x,y)\frac{\nabla\phi}{|\nabla\phi|}\right) + \beta(c_2 - c_1)\left(I - \frac{c_1+c_2}{2}\right)\right]|\nabla\phi|.$$

Efficient solutions for this equation can use either AOS [18, 19, 29] or ADI methods, coupled with a narrow band approach [7, 1], as first introduced in [12] for the geodesic active contour. In the next section we use a simple first order implicit alternating direction multiplicative scheme.

The following table summarizes the functionals, the resulting first variation for each functional, and the level set formulations for these terms.

Measure	$E(C)$	$\delta E/\delta C$	level set form
Weighted Area	$\iint_{\Omega_C} f(x,y)dxdy$	$-f(x,y)\vec{n}$	$-f(x,y)\|\nabla\phi\|$
Minimal Variance	$\iint_{\Omega_C}(I-c_1)^2dxdy+$ $\iint_{\Omega\setminus\Omega_C}(I-c_2)^2dxdy$	$(c_2-c_1)\times$ $(I-(c_1+c_2)/2)\vec{n}$	$(c_2-c_1)\times$ $(I-(c_1+c_2)/2)\|\nabla\phi\|$
GAC	$\oint_C g(C(s))ds$	$(\langle\nabla g,\vec{n}\rangle - \kappa g)\vec{n}$	$-\text{div}\left(g\frac{\nabla\phi}{\|\nabla\phi\|}\right)\|\nabla\phi\|$
Alignment	$\oint \|\langle\nabla I,\vec{n}\rangle\|ds$	$\text{sign}(\langle\nabla I,\vec{n}\rangle)\Delta I\vec{n}$	$\text{sign}(\langle\nabla I,\nabla\phi\rangle)\Delta I\|\nabla\phi\|$

6 Efficient Numerical Schemes

In [29] Weickert et al. used an unconditionally stable, and thus efficient, numerical scheme for non-linear isotropic image diffusion known as *additive operator splitting* (AOS), that was first introduced in [18, 19], and has some nice symmetry and parallel properties. Goldenberg et al. [12] coupled the AOS with Sethian's fast marching on regular grids [25] (see [27, 6] for related approaches), with multi-resolution [24], and with the narrow band approach [7, 1], as part of their fast geodesic active contour model for segmentation and object tracking in video sequences. Here, motivated by these results, we extend the efficient numerical methods for the geodesic active contour [4] presented in [12], for the variational edge integration models introduced in [16, 17], and the minimal variance [5]. We review

efficient numerical schemes and modify them in order to solve the level set formulation of edge integration and object segmentation problem in image analysis.

Let us analyze the following level set formulation,

$$\begin{aligned}\phi_t &= \left(\alpha \mathrm{div}\left(g(x,y)\frac{\nabla\phi}{|\nabla\phi|}\right) + \eta(\phi,\nabla I)\right)|\nabla\phi|,\\ \eta(\phi,\nabla I) &= \mathrm{sign}(\langle\nabla I,\nabla\phi\rangle)\Delta I + \beta(c_2-c_1)\left(I-\frac{c_1+c_2}{2}\right).\end{aligned}$$

If we assume $\phi(x,y;t)$ to be a distance function of its zero set, then, we could approximate the short time evolution of the above equation by setting $|\nabla\phi| = 1$. The short time evolution for a distance function ϕ is thereby

$$\begin{aligned}\phi_t &= \alpha \mathrm{div}\left(g(x,y)\nabla\phi\right) + \eta(\phi,\nabla I)\\ &= \alpha\frac{\partial}{\partial x}\left(g(x,y)\frac{\partial}{\partial x}\phi\right) + \alpha\frac{\partial}{\partial y}\left(g(x,y)\frac{\partial}{\partial y}\phi\right) + \eta(\phi,\nabla I).\end{aligned}$$

Note, that when using a narrow band around the zero set to reduce computational complexity on sequential computers, the distance from the zero set needs to be recomputed in order to determine the width of the band at each iteration. Thus, there is no additional computational cost in simplifying the model while considering a distance map rather than an arbitrary smooth function. We thereby enjoy both the efficiency of the simplified almost linear model, and the low computational cost of the narrow band.

Denote the operators

$$\begin{aligned}A_1 &= \frac{\partial}{\partial x}g(x,y)\frac{\partial}{\partial x}\\ A_2 &= \frac{\partial}{\partial y}g(x,y)\frac{\partial}{\partial y}.\end{aligned}$$

Using these notations we can write the evolution equation as

$$\phi_t = \alpha\left(A_1+A_2\right)\phi + \eta(\phi,\nabla I).$$

Next, we approximate the time derivative using forward approximation $\phi_t \approx \frac{\phi^{n+1}-\phi^n}{\tau}$, that yields the explicit scheme

$$\begin{aligned}\phi^{n+1} &= \phi^n + \tau\left(\alpha\left(A_1+A_2\right)\phi^n + \tau\eta(\phi^n,\nabla I)\right)\\ &= \left(\mathcal{I}+\tau\alpha(A_1+A_2)\right)\phi^n + \tau\eta(\phi^n,\nabla I),\end{aligned}$$

where, after sampling the x,y plane, $\mathcal{I}$ is the identity matrix and I is our input image. The operators A_l become matrix operators, and ϕ^n is represented as a vector in either column or row stack, depending on the acting operator. This way, the operators A_l are tri-diagonal, which makes its inverse computation fairly simple using Thomas algorithm. Note that

in the explicit scheme there is no need to invert any operator, yet the numerical time step is bounded for stability.

Let us first follow [29], and use a simple discretization for the A_l, $l \in \{1,2\}$ operators. For a given row, let

$$\frac{\partial}{\partial x}\left(g(x)\frac{\partial}{\partial x}\phi\right) \approx \sum_{j\in\mathcal{N}(i)} \frac{g_j+g_i}{2h^2}\left(\phi_j-\phi_i\right),$$

where $\mathcal{N}(i)$ is the set $\{i-1, i+1\}$, representing the two horizontal neighbors of pixel i, and h is the space between neighboring pixels. The elements of A_1 are thereby given by

$$a_{ij} = \begin{cases} \frac{g_i+g_j}{2h^2} & j \in \mathcal{N}(i) \\ -\sum_{k\in\mathcal{N}(i)} \frac{g_i+g_k}{2h^2} & j = i \\ 0 & \text{else.} \end{cases}$$

In order to construct an unconditionally stable scheme we use a locally one-dimensional (LOD) scheme adopted for our problem. We use the following scheme

$$\phi^{n+1} = \prod_{l=1}^{2} \left(\mathcal{I}-\tau\alpha A_l\right)^{-1}\left(\phi^n + \tau\eta(\phi^n, \nabla I)\right).$$

In one-dimension it is also known as fully implicit, or backward Euler scheme. It is a first order implicit numerical approximation, since we have that

$$\begin{aligned}(\mathcal{I}-\tau A_1)^{-1}(\mathcal{I}-\tau A_2)^{-1}(\phi+\tau\eta) &= \left(\mathcal{I}-\tau A_1-\tau A_2+O(\tau^2)\right)^{-1}(\phi+\tau\eta) \\ &= \phi+\tau(A_1+A_2)\phi+\tau\eta+O(\tau^2),\end{aligned}$$

where we applied the Taylor series expansion in the second equality. First order accuracy is sufficient, as our goal is the steady state solution. We also included the weighted region-balloon, minimal variance, and the alignment terms within this implicit scheme, while keeping the first order accuracy and stability properties of the method. The operators $\mathcal{I}-\tau A_l$ are positive definite symmetric operators, which make the implicit process unconditionally stable, using either the above multiplicative or the additive (AOS) schemes. If we have an indication that the contour is getting closer to its final destination, we could switch to an explicit scheme for the final refinement steps with a small time step. In this case, the time step should be within the stability limits of the explicit scheme. In our implementation we also use a multi–resolution pyramidal approach, where the coarse grid still captures the details of the objects we would like to detect.

FIGURE 4. Test images left to right: numbers with a small amount of noise, numbers with a large amount of noise, and a number with non-uniform illumination.

7 Examples

Figure 4 presents a set of simple input images in which the goal is to segment the numbers from the background. In all examples we used the image frame as initial condition for the active contour model, and applied a multi-resolution coarse to fine procedure, as in [24, 12], to speed up the segmentation process.

Figure 5 shows the segmentation result of the active contour model in the first case. In this example, the alignment and minimal variance terms were tuned to play the dominant role in the segmentation. The right frame, here and in the rest of the examples, shows the final contour in black painted on the original image in which the dynamic range is mapped into a sub-interval of the original one.

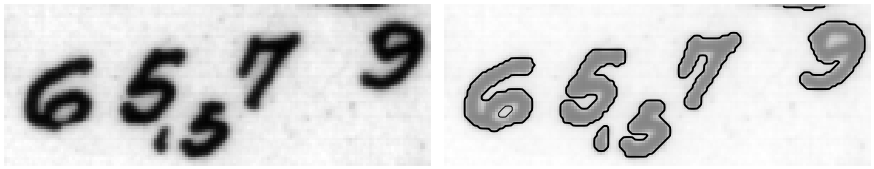

FIGURE 5. The simplest case in which alignment and minimal variance played the dominant factors in finding the exact location of the edges.

In the next example, shown in Fig. 6, high noise and uniform illumination calls for minimal variance coupled with regularization of the contour. The alignment term does not contribute much in such a high noise.

The last example demonstrates the effect of a non-uniform illumination. In this case, the minimal variance would be of little help, while the alignment term leads the active contour to the accurate edge locations, see Fig. 7.

In the appendix we present an oversimplified matlab code for the whole process. Note that in order to keep the program simple the multi-resolution is not presented, and the redistancing procedure was only roughly approximated, see [13] for more details.

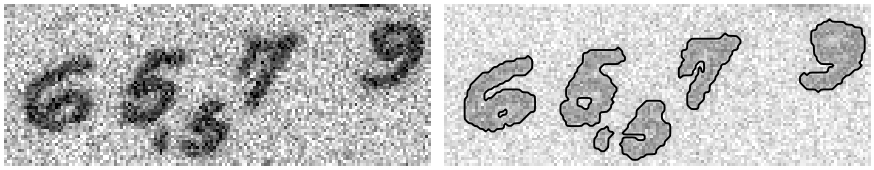

FIGURE 6. For noisy images the alignment term is turned off, while the minimal variance and regularization by the geodesic active contour are the important factors.

8 Conclusions

We explored fundamental geometric variational measures for edge integration and uniform region segmentation. Next, we computed the resulting first variation for these integral measures. The level set formulation for the resulting curve evolution equations was then implemented by efficient numerical schemes.

We are still far from the end of the road of implementing low level vision operators. Good numerical schemes for so-called 'solved' problems would probably change the way we process and analyze images in the future. Simple operations we take for granted, like edge detection and shape reconstruction, should be revisited and refined for the best possible solution. The exploration of the basic image analysis tools would improve our understanding of these processes and enable faster progress of feature extraction, learning, and classification procedures. Our philosophy of geometric variational interpretation for fundamental low level image analysis operators seem to be one promising direction for improving existing tools and designing new ones.

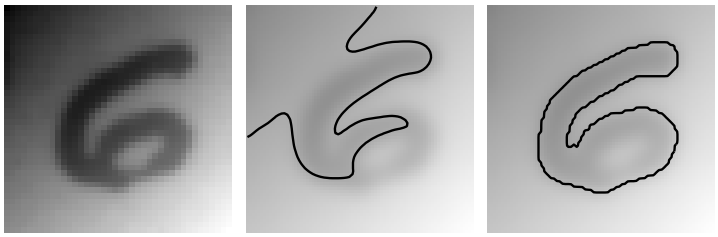

FIGURE 7. For non-uniform illumination without noise. The minimal variance term is turned off, while the alignment term plays the leading role in a good segmentation of the number from the background. Left: input image. Middle: Segmentation by threshold yields a meaningless result. Right: Active contour segmentation with a dominant alignment term.

Acknowledgments

Stimulating discussions with Alfred Bruckstein, Moshe Israeli, Irad Yavnhe, Roman Goldenberg, Yossi Rubner, and Alexander Brook at the Technion are gratefully acknowledged.

Appendix

```
%-------------------------------------------------------------------
% Locally One-Dimensional (LOD) implicit scheme for closed geometric
% active contour model+minimal variation+GAC+robust alignment.
% I = Input image matrix, Phi = Initial contour matrix (implicit form).
% Balloon = Weighted area factor (scalar)
% Align = Alignment force factor (scalar)
% Max_Lloyd = The Max-Lloyd/Chan-Vese threshold factor (scalar)
% k = The time step (tau scalar)
% iter = Maximal number of iterations (scalar)
function Phi=LOD_Active_Contour(I,Phi,Balloon,Align,Max_Lloyd,k,iter)
D2I = Dxx(I)+Dyy(I); P = Dx(I); Q=Dy(I);
g = 1./sqrt(1+P.^2+Q.^2); % example for computing g(x,y)
delta = 2; count=1;
while and(delta >0.0001,count<iter) %check if converged
    Phi_old = Phi;
    threshold = LloydMax(Phi,I); % Max-Lloyd/Chan-Vese term
    alignment = -sign(P.*Dx(Phi)+Q.*Dy(Phi)).*D2I; % Laplacian term
    Phi = Phi+k*(Balloon*g+Align*alignment+Max_Lloyd*threshold);
    for i=1:2, % i=1 => (I-tau*Ax) i=2 => (I-tau*Ay)
        Phi = Implicit(g(:),k,Phi); % (1/(I-tau*Ai))Phi
        Phi = Phi'; g = g'; % Transpose for Ay
    end % for i
    Phi = redistance(Phi); % Use fast marching for re-distancing
    delta = sum(sum((Phi_old-Phi).^2)) % Compute L2 norm
    count = count+1;
    imshow(I,[]); hold on; contour(Phi,[0 0],'r');hold off; drawnow;
end % while and function
```

```
%-------------------------------------------------------------------------
% Compute (1/(I- k*Al))Phi using Thomas algorithm
% k=time step, g=g(x,y) in column stack form
function u = Implicit(g,k,Phi)
gm = -k.*( g + g([end 1:end-1]))/2; % lower diag
gc = 1-k.*(-2*g - g([end 1:end-1]) -g([2:end 1]))/2; % main diag
gp = -k.*( g + g([2:end 1]))/2; % upper diag
u = Thomas(gc,gp(1:end-1),gm(2:end),Phi(:));
u = reshape(u,size(Phi));
%-------------------------------------------------------------------------
% Compute the Lloyd-Max/Chan-Vese thresholding
function force = LloydMax(Phi,I)
mask_in = (Phi<0); % inside the contour
mask_out = 1-mask_in; % rest of the domain
I_in = sum(sum(mask_in.*I))/sum(mask_in(:)); % mean value
I_out = sum(sum(mask_out.*I))/sum(mask_out(:));
force = (I_out-I_in).*(I-(I_in+I_out)/2);
%-------------------------------------------------------------------------
% 'Roughly' correct Phi to be a distance map of its zero set
function u=redistance(Phi);
u = (sign(Phi)+1)*999999; % set to infinity all positive
for i=1:2,
    l2 = 2;
    if i>1 u=(1-sign(Phi))*999999; end % set to infinity all negative
    while l2 > 1,
        v = Update(u,1);
        l2 = sum(sum((u-v).^2));
        u = v;
    end % while
    if i>1 u=u-up; else up=u; end %if
end % for
%-------------------------------------------------------------------------
% Solve |grad u|=F(x,y) parallel version of the FMM
function res = Update(u,F)
mx = min(u([2:end end],:), u([1 1:end-1],:));
my = min(u(:,[2:end end]), u(:,[1 1:end-1]));
delm = (mx -my);
mask = (abs(delm) < F);
res= min(mask.* (mx+my+sqrt(2.*F.^2- delm .^2))./2 + ...
    (1-mask).* (min(mx,my)+F) , u);
```

```
%-------------------------------------------------------------------
function f = Dmx(P)
f = P - P([1 1:end-1],:);
%-------------------------------------------------------------------
function f = Dpx(P)
f = P([2:end end],:) - P;
%-------------------------------------------------------------------
function f = Dx(P)
f = (Dpx(P)+Dmx(P))/2;
%-------------------------------------------------------------------
function f = Dy(P)
f = (Dx(P'))';
%-------------------------------------------------------------------
function f = Dxx(P)
f = Dpx(P)-Dmx(P);
%-------------------------------------------------------------------
function f = Dyy(P)
f = (Dxx(P'))';
%-------------------------------------------------------------------
% Thomas Algorithm for trilinear diagonally dominant system:
% B u = d (solve for u); B is given by its 3 diagonals:
% alpha(1:N)=main diagonal, beta(1:N-1)=upper diagonal,
% gamma(1:N-1) = lower diagonal, (Compile first!)
function u = Thomas(alpha,beta,gamma,d)
N = length(alpha); r = beta;
l = zeros(N-1,1); u = zeros(N,1);
m = u; % zero
m(1) = alpha(1);
for i=1:N-1,
    l(i) = gamma(i)/m(i);
    m(i+1) = alpha(i+1)-l(i)*beta(i);
end % for
y = u; % zero
y(1) = d(1);
for i = 2:N,
    y(i) = d(i)-l(i-1)*y(i-1);
end % for
u(N) = y(N)/m(N);
for i = N-1:-1:1,
    u(i) = (y(i)-beta(i)*u(i+1))/m(i);
end % for
```

9 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] J. Canny. A computational approach to edge detection. *IEEE Trans. on PAMI*, 8(6):679–698, 1986.

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

[4] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.

[5] T. Chan and L. Vese. An active contour model without edges. In *Scale-Space Theories in Computer Vision*, pages 141–151, 1999.

[6] 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.

[7] D. L. Chopp. Computing minimal surfaces via level set curvature flow. Ph.D Thesis, Lawrence Berkeley Lab. and Dep. of Math. LBL-30685, Uni. of CA. Berkeley, May 1991.

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

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

[10] P. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980.

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

[12] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Fast geodesic active contours. *IEEE Tran. on Image Processing*, 10(10):1467–1475, 2001.

[13] R. Goldenberg, R. K. E. Rivlin, and M. Rudzsky. Cortex segmentation - a fast variational geometric approach. In *IEEE Workshop on Variational and Level Set Methods in Computer Vision*, Vancouver, Canada, July 2001.

[14] R. Haralick. Digital step edges from zero crossing of second directional derivatives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(1):58–68, January 1984.

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

[16] R. Kimmel and A. M. Bruckstein. Regularized Laplacian zero crossings as optimal edge integrators. In *Proceedings of Image and Vision Computing, IVCNZ01*, New Zealand, November 2001.

[17] R. Kimmel and A. M. Bruckstein. On edge detection edge integration and geometric active contours. In *Proceedings of Int. Symposium on Mathematical Morphology, ISMM2002*, Sydney, New South Wales, Australia, April 2002.

[18] T. Lu, P. Neittaanmaki, and X.-C. Tai. A parallel splitting up method and its application to Navier-Stokes equations. *Applied Mathematics Letters*, 4(2):25–29, 1991.

[19] T. Lu, P. Neittaanmaki, and X.-C. Tai. A parallel splitting up method for partial differential equations and its application to Navier-Stokes equations. *RAIRO Math. Model. and Numer. Anal.*, 26(6):673–708, 1992.

[20] 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.

[21] D. Marr. *Vision*. Freeman, San Francisco, 1982.

[22] D. Marr and E. Hildreth. Theory of edge detection. *Proc. of the Royal Society London B*, 207:187–217, 1980.

[23] 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.

[24] N. Paragios and R. Deriche. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Trans. on PAMI*, 22(3):266–280, 2000.

[25] J. A. Sethian. A review of the theory, algorithms, and applications of level set methods for propagating interfaces. *Acta Numerica*, Cambridge University Press, 1995.

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

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

[28] A. Vasilevskiy and K. Siddiqi. Flux maximizing geometric flows. In *Proceedings ICCV'01*, Vancouver, Canada, July 2001.

[29] 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.

[30] S. Zhu, T. Lee, and A. Yuille. Region competition: Unifying snakes, region growing, energy/bayes/mdl for multi-band image segmentation. In *Proc. ICCV'95*, pages 416–423, Cambridge, June 1995.