Published in Image Processing On Line on 2012–08–08.
Submitted on 2012–00–00, accepted on 2012–00–00.
ISSN 2105–1232 © 2012 IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
http://dx.doi.org/10.5201/ipol.2012.g-cv

Chan–Vese Segmentation

Pascal Getreuer

Yale University (pascal.getreuer@yale.edu)

Abstract

While many segmentation methods rely heavily in some way on edge detection, the "Active Contours Without Edges" method by Chan and Vese [7, 9] ignores edges completely. Instead, the method optimally fits a two-phase piecewise constant model to the given image. The segmentation boundary is represented implicitly with a level set function, which allows the segmentation to handle topological changes more easily than explicit snake methods.
This article describes the level set formulation of the Chan-Vese model and its numerical solution using a semi-implicit gradient descent. We also discuss the Chan–Sandberg–Vese method [8], a straightforward extension of Chan–Vese for vector-valued images.

Source Code

ANSI C source code to produce the same results as the demo is accessible on the article web page http://dx.doi.org/10.5201/ipol.2012.g-cv. Future software releases and updates will be posted at http://dev.ipol.im/~getreuer/code.

Supplementary Material

A video on the article web page http://dx.doi.org/10.5201/ipol.2012.g-cv shows animated segmentation evolutions for several of the examples in this work.

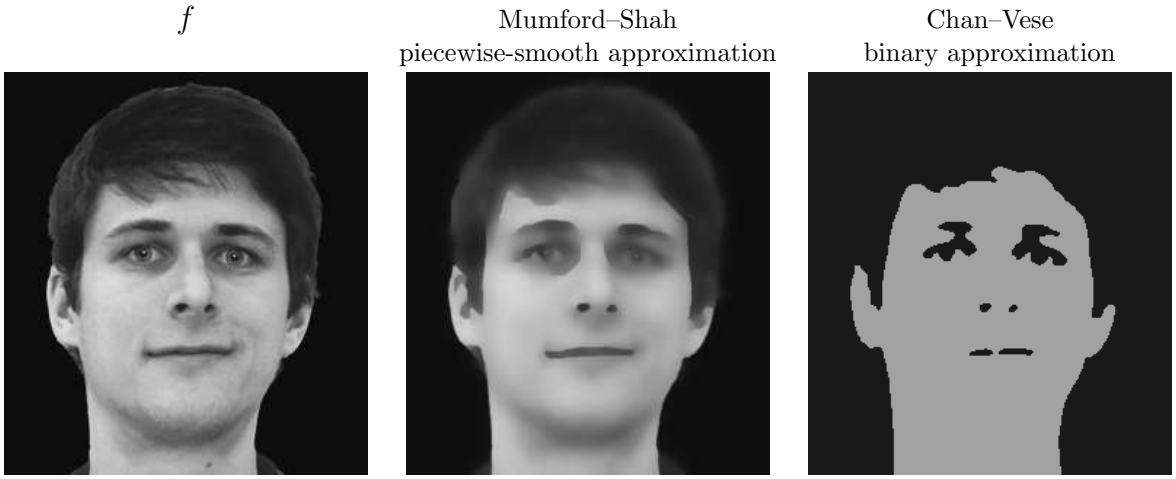
Keywords: image segmentation, level sets

1 Simplified Mumford–Shah Model

Let f denote the given grayscale image on a domain Ω to be segmented. The Chan–Vese method is inspired by the Mumford–Shah model. Mumford and Shah [3] approximate the image f by a piecewise-smooth function u as the solution of the minimization problem

$$\arg\min_{u,C} \ \mu\,\mathrm{Length}(C) + \lambda \int_\Omega \big(f(x) - u(x)\big)^2\,dx + \int_{\Omega\setminus C} |\nabla u(x)|^2\,dx, \tag{1}$$

where C is an edge set curve where u is allowed to be discontinuous. The first term ensures regularity of C, the second term encourages u to be close to f, and the third term ensures that u is differentiable on $\Omega\backslash C$. The Mumford–Shah approximation suggests selecting this edge set C as the segmentation boundary. The following example illustrates the difference between the Mumford–Shah and Chan–Vese function models. The Mumford–Shah solution was computed using the reduced Ambrosio–Tortorelli method of Vese and Chan [6].



Although it is a natural way to pose segmentation, algorithms for solving the general Mumford–Shah model tend to be relatively complicated and computationally expensive. As a simplification, Mumford and Shah also considered a piecewise constant formulation,

$$\arg\min_{u,C} \ \mu\,\mathrm{Length}(C) + \int_\Omega \big(f(x) - u(x)\big)^2\,dx, \tag{2}$$

where u is required to be constant on each connected component of $\Omega\backslash C$. In this case, C is necessarily the boundary of a closed set, i.e., C is composed of closed curves. The minimization problem is nonconvex. Existence of a solution of the piecewise constant model was proved by Mumford and Shah [3]. A much simpler proof of existence is given by Morel and Solimini [4].

Compared to the piecewise constant Mumford–Shah model, the key differences with the Chan–Vese model are an additional term penalizing the enclosed area and a further simplification that u is allowed to have only two values,

$$u(x) = \begin{cases} c_1 & \text{where } x \text{ is inside } C, \\ c_2 & \text{where } x \text{ is outside } C, \end{cases} \tag{3}$$

where C is the boundary of a closed set and c_1, c_2 are the values of u respectively inside and outside of C. The Chan–Vese method is to find among all u of this form the one that best approximates f,

$$\begin{aligned} \arg\min_{c_1,c_2,C} \ & \mu\,\mathrm{Length}(C) + \nu\,\mathrm{Area}(\mathit{inside}(C)) \\ & + \lambda_1 \int_{\mathit{inside}(C)} |f(x) - c_1|^2\,dx + \lambda_2 \int_{\mathit{outside}(C)} |f(x) - c_2|^2\,dx. \end{aligned} \tag{4}$$

The first term controls the regularity by penalizing the length. The second term penalizes the enclosed area of C to control its size. The third and fourth terms penalize discrepancy between the piecewise constant model u and the input image f. By finding a local minimizer of this problem, a segmentation is obtained as the best two-phase piecewise constant approximation u of the image f.

A proof of existence and regularity for the Chan–Vese model can be trivially deduced from the proof of existence and regularity for the piecewise constant model by Morel and Solimini [4].

2 Level Set Functions

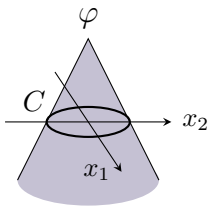
The minimization problem requires minimizing over all set boundaries C. This is accomplished by applying the level set technique introduced by Osher and Sethian [2]. Instead of manipulating C explicitly, it is represented as the zero-crossing of a *level set function* φ by the relationship

$$C = \{x \in \Omega : \varphi(x) = 0\}. \tag{5}$$

Furthermore, the inside and outside of C are distinguished by the sign of φ. As an example,

$$\varphi(x) = r - \sqrt{x_1^2 + x_2^2} \tag{6}$$

is a level set function for a circle of radius r.



Level set function for a circle of radius r.

Note that for a given C, there is more than one possible level set representation: if φ is a level set function for Ω, then so is any other function ψ having the same sign, $\operatorname{sign}(\psi(x)) = \operatorname{sign}(\varphi(x))$ for all x.

Provided that φ is smooth enough and that φ is indeed a distance function (i.e., $|\varphi| = 1$), then the Chan–Vese minimization can be equivalently rewritten in terms of the level set function φ as

$$\begin{aligned}\operatorname*{arg\,min}_{c_1,c_2,\varphi} \ & \mu \int_\Omega \delta\big(\varphi(x)\big)|\nabla\varphi(x)|\,dx + \nu \int_\Omega H\big(\varphi(x)\big)\,dx \\ & + \lambda_1 \int_\Omega |f(x) - c_1|^2 H\big(\varphi(x)\big)\,dx + \lambda_2 \int_\Omega |f(x) - c_2|^2 \big(1 - H\big(\varphi(x)\big)\big)\,dx,\end{aligned} \tag{7}$$

where H denotes the Heaviside function and δ the Dirac mass, its distributional derivative,

$$H(t) = \begin{cases} 1 & t \geq 0, \\ 0 & t < 0, \end{cases} \quad \delta(t) = \frac{d}{dt} H(t). \tag{8}$$

Note that $H(\varphi)$ is the indicator function of the set enclosed by C. Through this connection, the area of Ω is obtained in the second integral as the integral of $H(\varphi)$. The first integral containing the Dirac mass δ is not really an integral, but a curve integral along the boundary C. Provided φ is a distance function, the length of C is obtained as the total variation of $H(\varphi)$,

$$\operatorname{Length}(C) = \int_\Omega \big|\nabla H\big(\varphi(x)\big)\big|\,dx = \int_\Omega \delta\big(\varphi(x)\big)|\nabla\varphi(x)|\,dx. \tag{9}$$

The minimization is solved by alternatingly updating c_1, c_2 and φ. For fixed φ, the optimal values of c_1 and c_2 are the region averages

$$c_1 = \frac{\int_\Omega f(x) H\big(\varphi(x)\big)\,dx}{\int_\Omega H\big(\varphi(x)\big)\,dx}, \tag{10}$$

$$c_2 = \frac{\int_\Omega f(x)\big(1 - H\big(\varphi(x)\big)\big)\,dx}{\int_\Omega \big(1 - H\big(\varphi(x)\big)\big)\,dx}. \tag{11}$$

For the minimization with respect to φ, H is regularized as

$$H_\epsilon(t) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{t}{\epsilon}\right)\right). \tag{12}$$

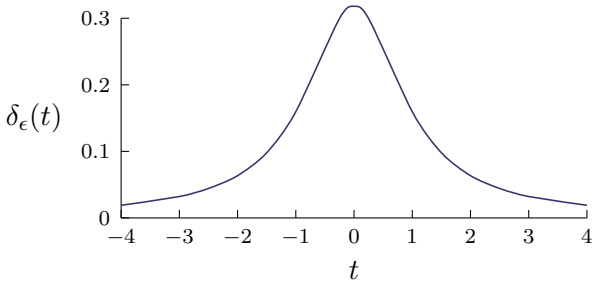
and δ_ϵ is its derivative,

$$\delta_\epsilon(t) := \frac{d}{dt} H_\epsilon(t) = \frac{\epsilon}{\pi(\epsilon^2 + t^2)}. \tag{13}$$

In the implementation, $\epsilon = 1$. For fixed c_1 and c_2, gradient descent with respect to φ is

$$\begin{cases} \dfrac{\partial \varphi}{\partial t} = \delta_\epsilon(\varphi)\Big[\mu \operatorname{div}\left(\dfrac{\nabla \varphi}{|\nabla \varphi|}\right) - \nu - \lambda_1 (f - c_1)^2 + \lambda_2 (f - c_2)^2\Big] & \text{in } \Omega, \\ \dfrac{\delta_\epsilon(\varphi)}{|\nabla \varphi|} \dfrac{\partial \varphi}{\partial \vec{n}} = 0 \quad \text{on } \partial\Omega, \end{cases} \tag{14}$$

where $\vec{n}$ is the outward normal on the image boundary.



Regularized delta $\delta_\epsilon(t)$ with $\epsilon = 1$.

For the initialization, an effective choice is the function

$$\varphi(x) = \sin(\tfrac{\pi}{5} x_1) \sin(\tfrac{\pi}{5} y) \tag{15}$$

which defines the initial segmentation as a checkerboard shape. This initialization has been observed to have fast convergence. When using this initialization with $\nu = 0$, the meaning of the result's "inside" vs. "outside" are arbitrary.

Alternatively, one may specify a contour C for the initialization. A level set representation of C can be constructed as

$$\varphi(x) = \begin{cases} +1 & \text{where } x \text{ is inside } C, \\ -1 & \text{where } x \text{ is outside } C. \end{cases} \tag{16}$$

3 Numerical Implementation

Here we describe the semi-implicit gradient descent for solving the Chan–Vese minimization as developed in the original work [7, 9]. This is not the only way to solve the problem, for example, He and Osher [14] introduced an algorithm based on the topological derivative, Badshah and Chen [16] developed a multigrid method, and Zehiry, Xu, and Sahoo [15] and Bae and Tai [17] proposed fast algorithms based on graph cuts.

For the numerical implementation, suppose that f is sampled on a regular grid $\Omega = \{0, \dots, M\} \times \{0, \dots, M\}$. The evolution of φ is discretized in space according to

$$\begin{aligned} \frac{\partial \varphi_{i,j}}{\partial t} = \delta_\epsilon(\varphi_{i,j}) \Bigg[\mu \Bigg(\nabla_x^- \frac{\nabla_x^+ \varphi_{i,j}}{\sqrt{\eta^2 + (\nabla_x^+ \varphi_{i,j})^2 + (\nabla_y^0 \varphi_{i,j})^2}} + \nabla_y^- \frac{\nabla_y^+ \varphi_{i,j}}{\sqrt{\eta^2 + (\nabla_x^0 \varphi_{i,j})^2 + (\nabla_y^+ \varphi_{i,j})^2}} \Bigg) \\ - \nu - \lambda_1 (f_{i,j} - c_1)^2 + \lambda_2 (f_{i,j} - c_2)^2 \Bigg], \quad i, j = 1, \dots, M - 1, \end{aligned} \tag{17}$$

where ∇_x^+ denotes forward difference in the x dimension, ∇_x^- denotes backward difference, and $\nabla_x^0 := (\nabla_x^+ + \nabla_x^-)/2$ is central difference, and similarly in the y dimension. The curvature is regularized by including a small positive parameter η in the denominators that prevents division by zero. In the implementation, $\eta = 10^{-8}$. Let

$$A_{i,j} = \frac{\mu}{\sqrt{\eta^2 + (\nabla_x^+\varphi_{i,j})^2 + (\nabla_y^0\varphi_{i,j})^2}}, \quad B_{i,j} = \frac{\mu}{\sqrt{\eta^2 + (\nabla_x^0\varphi_{i,j})^2 + (\nabla_y^+\varphi_{i,j})^2}}, \tag{18}$$

then the discretization can be written as

$$\begin{aligned}\frac{\partial\varphi_{i,j}}{\partial t} = \delta_\epsilon(\varphi_{i,j})\Big[&\big(A_{i,j}(\varphi_{i+1,j} - \varphi_{i,j}) - A_{i-1,j}(\varphi_{i,j} - \varphi_{i-1,j})\big)\\ &+ \big(B_{i,j}(\varphi_{i,j+1} - \varphi_{i,j}) - B_{i,j-1}(\varphi_{i,j} - \varphi_{i,j-1})\big)\\ &- \nu - \lambda_1(f_{i,j} - c_1)^2 + \lambda_2(f_{i,j} - c_2)^2\Big].\end{aligned} \tag{19}$$

On the right-hand side, the first two terms discretize the curvature,

$$\begin{aligned}\operatorname{div}\left(\frac{\nabla\varphi}{|\nabla\varphi|}\right) &= \partial_x\left(\frac{\partial_x\varphi}{\sqrt{(\partial_x\varphi)^2 + (\partial_y\varphi)^2}}\right) + \partial_y\left(\frac{\partial_y\varphi}{\sqrt{(\partial_x\varphi)^2 + (\partial_y\varphi)^2}}\right)\\ &\approx \nabla_x^-\frac{\nabla_x^+\varphi}{\sqrt{\eta^2 + (\nabla_x^+\varphi)^2 + (\nabla_y^0\varphi)^2}} + \nabla_y^-\frac{\nabla_y^+\varphi}{\sqrt{\eta^2 + (\nabla_x^0\varphi)^2 + (\nabla_y^+\varphi)^2}}.\end{aligned} \tag{20}$$

The reason for the mix of various differences is so that combined result is centered yet better localized than simply using centered differences for each derivative. In the first term, the numerator $\nabla_x^+\varphi_{i,j}$ is logically centered at $(i+\frac{1}{2}, j)$. The denominator is selected as $(\nabla_x^+\varphi)^2 + (\nabla_y^0\varphi)^2$ so that it, and hence the quotient, are also centered at $(i+\frac{1}{2}, j)$. By applying the backward difference ∇_x^- to the quotient, the result is centered at (i, j) as desired. The choice of differences is analogous in the second term along the y coordinate.

Time is discretized with a semi-implicit Gauss–Seidel method [5]. The strategy is such that only one copy of the φ array needs to be stored in memory. The values $\varphi_{i,j}$ are updated in-place as they are computed.

The values $\varphi_{i,j}$, $\varphi_{i-1,j}$, $\varphi_{i,j-1}$ are evaluated at timestep $n+1$ and all others at timestep n,

$$\begin{aligned}\frac{\varphi_{i,j}^{n+1} - \varphi_{i,j}^n}{dt} = \delta_\epsilon(\varphi_{i,j}^n)\big[&A_{i,j}\varphi_{i+1,j}^n + A_{i-1,j}\varphi_{i-1,j}^{n+1} + B_{i,j}\varphi_{i,j+1}^n + B_{i,j-1}\varphi_{i,j-1}^{n+1}\\ &- (A_{i,j} + A_{i-1,j} + B_{i,j} + B_{i,j-1})\varphi_{i,j}^{n+1}\\ &- \nu - \lambda_1(f_{i,j} - c_1)^2 + \lambda_2(f_{i,j} - c_2)^2\big].\end{aligned} \tag{21}$$

This allows φ at timestep $n+1$ to be solved by one Gauss–Seidel sweep from left to right, top to bottom,

$$\begin{aligned}\varphi_{i,j}^{n+1} \leftarrow \big[&\varphi_{i,j}^n + dt\,\delta_\epsilon(\varphi_{i,j}^n)\big(A_{i,j}\varphi_{i+1,j}^n + A_{i-1,j}\varphi_{i-1,j}^{n+1} + B_{i,j}\varphi_{i,j+1}^n + B_{i,j-1}\varphi_{i,j-1}^{n+1}\\ &- \nu - \lambda_1(f_{i,j} - c_1)^2 + \lambda_2(f_{i,j} - c_2)^2\big)\big]\\ &/\ \big[1 + dt\,\delta_\epsilon(\varphi_{i,j})(A_{i,j} + A_{i-1,j} + B_{i,j} + B_{i,j-1})\big].\end{aligned} \tag{22}$$

The coefficients A and B are computed using the latest available values of φ,

$$A_{i,j} = \frac{\mu}{\sqrt{\eta^2 + (\varphi_{i+1,j}^n - \varphi_{i,j}^n)^2 + ((\varphi_{i,j+1}^n - \varphi_{i,j-1}^{n+1})/2)^2}}, \tag{23}$$

$$B_{i,j} = \frac{\mu}{\sqrt{\eta^2 + ((\varphi_{i+1,j}^n - \varphi_{i-1,j}^{n+1})/2)^2 + (\varphi_{i,j}^n - \varphi_{i+1,j}^n)^2}}. \tag{24}$$

The boundary condition is enforced by duplicating pixels near the borders,

$$\varphi_{-1,j} = \varphi_{0,j}, \quad \varphi_{M,j} = \varphi_{M-1,j}, \quad \varphi_{i,-1} = \varphi_{i,0}, \quad \varphi_{i,M} = \varphi_{i,M-1}. \tag{25}$$

Optionally, the level set function can be reinitialized after every N iterations by replacing φ with the signed distance function to C or any other function having the same sign at each point. This change does not modify segmentation boundary itself, but it prevents new components from appearing far away from the current boundary.

To terminate the method, one approach is to stop when the L^2 difference between φ^{n+1} and φ^n is below a tolerance. In the implementation, the default value of *tol* is 10^{-3}. The overall algorithm is, as described in the original paper [9],

Initialize φ
for $n = 1, 2, \ldots$ **do**
  Compute c_1 and c_2 as the region averages
  Evolve φ with one semi-implicit timestep
  if $\|\varphi^{n+1} - \varphi^n\|_2 / |\Omega| \leq tol$ **then stop**
  (Optional) If n is divisible by N, reinitialize φ

Algorithm 1

The computational cost per iteration is linear in the number of pixels. The required number of iterations depends greatly on the timestep dt and the initialization. The contour tends to evolve slowly if it has low curvature (e.g., a large ellipse) and requires possibly in the thousands of iterations to converge. An initialization with high curvature like the checkerboard initialization mentioned in the previous section tends to converge much faster.

4 Extension to Vector-Valued Data

The Chan–Sandberg–Vese model [8] is an extension to segmentation of vector-valued data,

$$\begin{aligned} \underset{c_1,c_2,C}{\arg\min} \;& \mu \operatorname{Length}(C) + \nu \operatorname{Area}(\mathit{inside}(C)) \\ &+ \lambda_1 \int_{\mathit{inside}(C)} \|f(x) - c_1\|^2 \, dx + \lambda_2 \int_{\mathit{outside}(C)} \|f(x) - c_2\|^2 \, dx. \end{aligned} \tag{26}$$

Here, the range of f is d-dimensional, and c_1 and c_2 are vectors of size d. As in the scalar case, c_1 and c_2 are the region averages of f inside and outside of C. In the case of color images, c_1 and c_2 are the average color values of the two segments. The source code included with this article implements Chan–Sandberg–Vese for any number of range dimensions.

The Chan–Sandberg–Vese problem is the same as the Chan–Vese minimization except that the absolute values $|f(x) - c_1|$ and $|f(x) - c_2|$ are replaced with the Euclidean vector norms $\|f(x) - c_1\|$ and $\|f(x) - c_2\|$. All image channels participate jointly in driving the segmentation process. The problem can be represented by a level set and discretized similarly, leading to the semi-implicit update of φ as

$$\begin{aligned} \frac{\varphi_{i,j}^{n+1} - \varphi_{i,j}^n}{dt} = \delta_\epsilon(\varphi_{i,j}^n) \Bigg[\mu \Bigg(\nabla_x^- \frac{\nabla_x^+ \varphi_{i,j}^{n+1}}{\sqrt{\eta^2 + (\nabla_x^+ \varphi_{i,j}^n)^2 + (\nabla_y^0 \varphi_{i,j}^n)^2}} + \nabla_y^- \frac{\nabla_y^+ \varphi_{i,j}^{n+1}}{\sqrt{\eta^2 + (\nabla_x^0 \varphi_{i,j}^n)^2 + (\nabla_y^+ \varphi_{i,j}^n)^2}} \Bigg) \\ - \nu - \lambda_1 \|f_{i,j} - c_1\|^2 + \lambda_2 \|f_{i,j} - c_2\|^2 \Bigg], \quad i, j = 1, \ldots, M-1, \end{aligned} \tag{27}$$

The segmentation is model is invariant under unitary transformations of the channels, e.g., performing the segmentation in the RGB color space is equivalent to using a color space that is related to RGB by a unitary transform. Thus the model is fairly insensitive to the choice of basis.

4.1 Other Extensions

Chan–Vese has also been extended for segmentation based on Gabor texture cues [12]. A disadvantage of Chan–Vese is that it only provides segmentation into two phases. Extensions of Chan–Vese have been developed for nested segmentation curves [13] and multiphase segmentation [11, 18]. Tsai, Yezzi, and Willsky [10] developed a general level set curve evolution framework similar to Chan–Vese for Mumford–Shah segmentation and simultaneous segmentation, denoising, and inpainting.

5 Examples

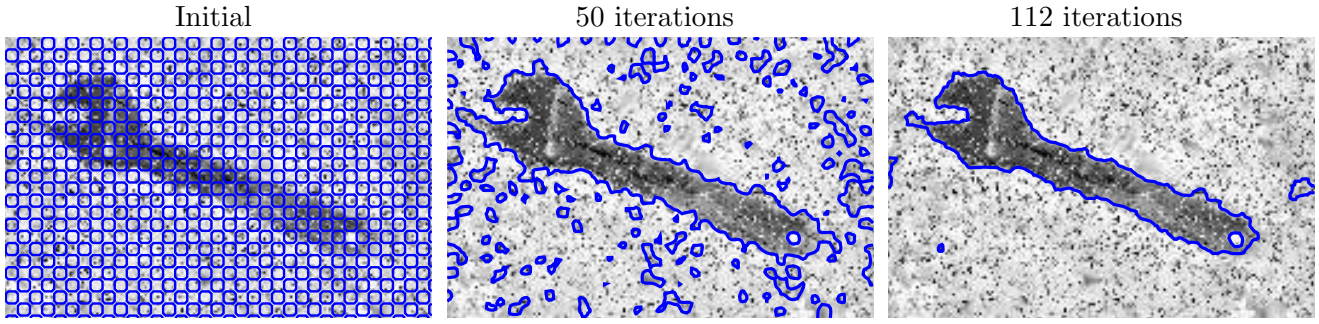
In the examples, we use (unless where specified otherwise) the default parameters length penalty $\mu = 0.2$, area penalty $\nu = 0$, fit weights $\lambda_1 = \lambda_2 = 1$, time step $dt = 0.5$, convergence tolerance $tol = 10^{-3}$, Heaviside regularization $\epsilon = 1$, curvature regularization $\eta = 10^{-8}$ and initialization of φ with the checkerboard function

$$\varphi(x) = \sin(\tfrac{\pi}{5}x_1)\sin(\tfrac{\pi}{5}x_2), \tag{28}$$

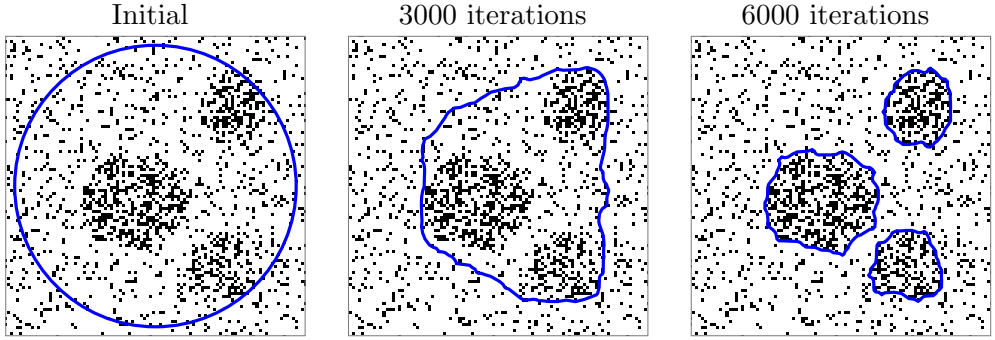
and reinitialization is not performed unless specified.

5.1 Grayscale Segmentation

The following example demonstrates Chan–Vese segmentation on a noisy image of a wrench (with the parameters stated above). The evolution converges in 112 iterations.

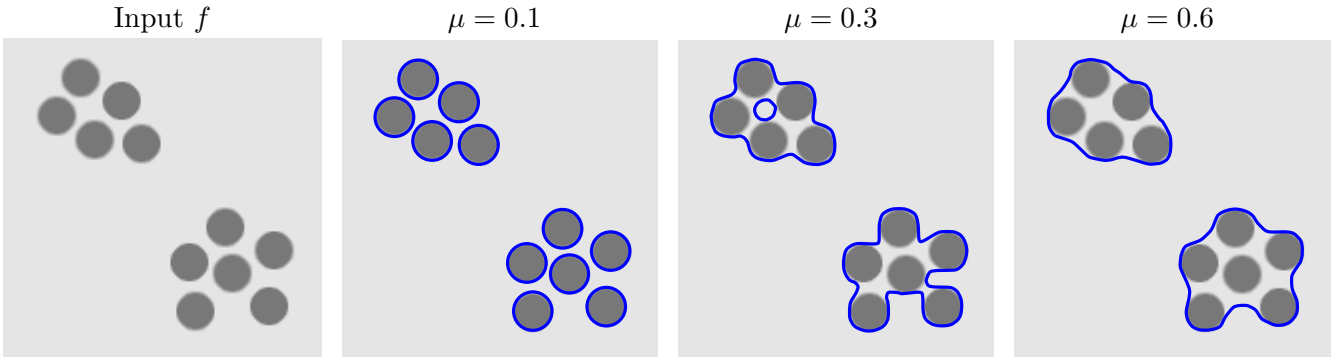


The next example highlights how the segmentation does not depend on image edges in an experiment similar to one of Chan and Vese's examples [9, Figure 10]. The method identifies the clusters from their darker average intensity, even though the cluster boundaries are not well defined. The initial boundary is a circle and $\mu = 0.3$.



5.2 Effect of Varying μ

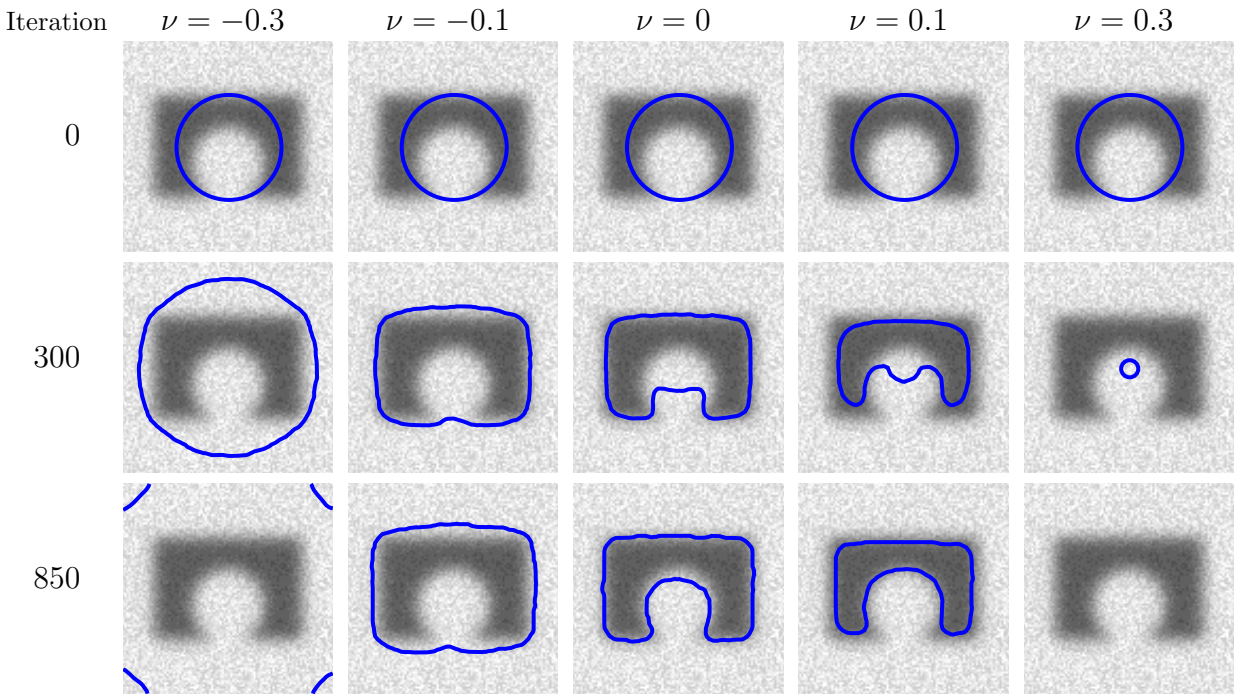
While Chan–Vese has quite a few tuning parameters, the most important is μ. Parameter μ adjusts the length penalty, which balances between fitting the input image more accurately (smaller μ) vs. producing a smoother boundary (larger μ). In the example below, there are two groups of circles. Depending on μ, the circles are either segmented individually or as two clusters.



Chan–Vese results with different values of μ.

5.3 Effect of Varying ν

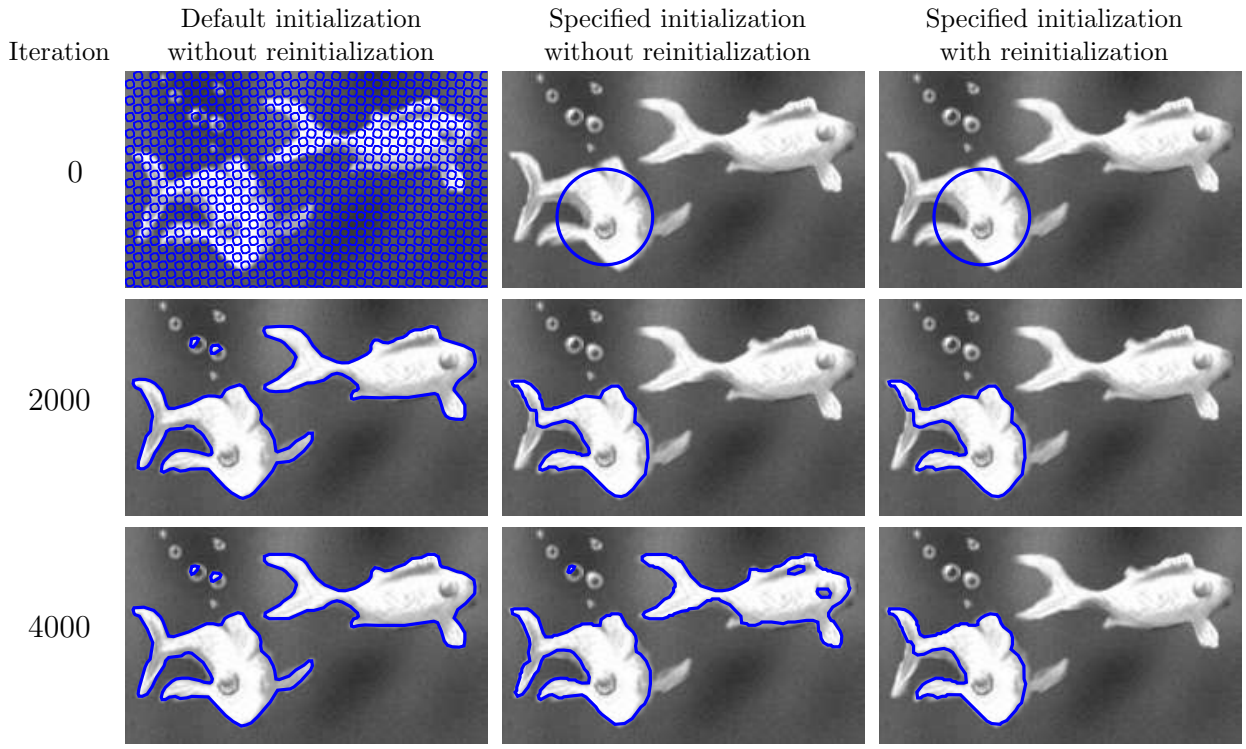
Parameter ν sets the penalty (or reward, if $\nu < 0$) for area inside C. Note that this parameter is meaningful only when there is a prescribed inside vs. outside of the segmentation boundary. In the figure below, the evolution is shown over 850 iterations with five different values of ν where the initialization is a circle. When ν is too negative, the boundary expands to fill the full domain, and when ν is too positive, the boundary shrinks until it vanishes.



5.4 Effect of the Initialization and Reinitialization

The Chan–Vese model generally has multiple local minimizers. By using an initial boundary, specific objects in the image can be segmented. Reinitialization may need to be used to avoid including

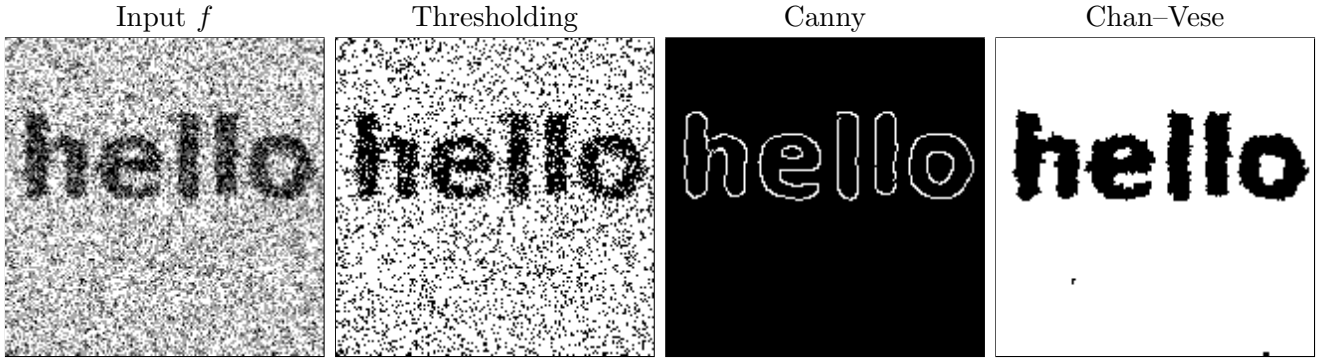
separate objects. The left-most column shows the evolution with the checkerboard initialization with $\mu = 0.15$. The other two columns show where the initialization is a circle on the left fish, both with and without using reinitialization, over 4000 iterations.



In the center column, a separate component appears to include the other fish. In the right image, reinitialization is performed after 2000 iterations, allowing it to maintain a single closed curve.

5.5 Comparison with Thresholding and Canny Edge Detection

Two standard methods for segmentation are thresholding according to the image intensity and the Canny edge detector [1]. In this example, we compare thresholding at the 50% graylevel, Canny edge detection with sensitivity threshold of 0.6 and filtering with a Gaussian with standard deviation 3, and Chan–Vese segmentation.

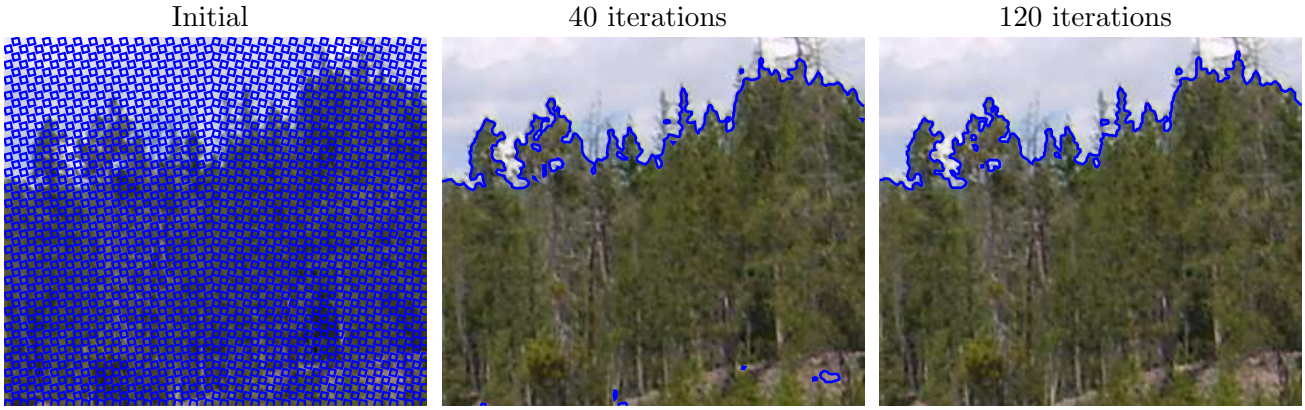


Results with different segmentation methods.

Direct thresholding is inadequate on noisy images like this one. The result with Canny edge detection is good, but curves often have small gaps and spurs. With Chan–Vese, the level set representation guarantees that the segmentation boundaries are always closed curves.

5.6 Color Image Segmentation

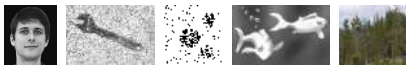
The Chan–Sandberg–Vese model can be used to segment color images. In this color image, Chan–Sandberg–Vese is applied in the RGB color space to separate the sky from the trees in 120 iterations.



Acknowledgments

This material is based upon work supported by the National Science Foundation under Award No. DMS-1004694. Work partially supported by the Office of Naval Research under grant N00014-97-1-0839 and by the European Research Council, advanced grant "Twelve labours."

Image Credits


Pascal Getreuer, CC-BY

References

[1] J. Canny, "A Computational Approach To Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 679–698, 1986. http://dx.doi.org/10.1109/TPAMI.1986.4767851

[2] S. Osher, J.A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations," Journal of Computational Physics, vol. 79, no. 1, pp. 12–49, 1988. http://dx.doi.org/10.1016/0021-9991(88)90002-2

[3] D. Mumford, J. Shah, "Optimal approximation by piecewise smooth functions and associated variational problems," Communications on Pure and Applied Mathematics, vol. 42, pp. 577–685, 1989. http://dx.doi.org/10.1002/cpa.3160420503

[4] J.M. Morel, S. Solimini, "Variational Models for Image Segmentation: with seven image processing experiments," Birkhäuser, 1994. ISBN: 0817637206.

[5] G. Aubert, L.A. Vese, "A variational method in image recovery," SIAM Journal on Numerical Analysis, vol. 34, no. 5, pp. 1948–1979, 1997. http://dx.doi.org/10.1137/S003614299529230X

[6] L.A. Vese and T.F. Chan, "Reduced non-convex functional approximations for image restoration & segmentation," UCLA CAM Reports 97-56, 1997. ftp://ftp.math.ucla.edu/pub/camreport/cam97-56.ps.gz

[7] T.F. Chan, L.A. Vese, "An active contour model without edges," Lecture Notes in Computer Science, vol. 1682, pp. 141–151, 1999. http://dx.doi.org/10.1007/3-540-48236-9_13

[8] T.F. Chan, B. Sandberg, L.A. Vese, "Active Contours Without Edges for Vector-Valued Images," Journal of Visual Communication and Image Representation, vol. 11, pp. 130–141, 2000. http://dx.doi.org/10.1006/jvci.1999.0442

[9] T.F. Chan, L.A. Vese, "Active Contours Without Edges," IEEE Transactions on Image Processing, vol. 10, no. 1, pp. 266–277, 2001. http://dx.doi.org/10.1109/83.902291

[10] A. Tsai, A. Yezzi, A.S. Willsky, "Curve Evolution Implementation of the Mumford Shah Functional for Image Segmentation, Denoising, Interpolation, and Magnification," IEEE Transactions on Image Processing, vol. 10, no. 8, pp. 1169–1186, 2001. http://dx.doi.org/10.1109/83.935033

[11] L.A. Vese, T.F. Chan, "A Multiphase Level Set Framework for Image Segmentation Using the Mumford and Shah Model," International Journal of Computer Vision, vol. 50, no. 3, pp. 271–293, 2002. http://dx.doi.org/10.1023/A:1020874308076

[12] B. Sandberg, T.F. Chan, L.A. Vese, "A Level-Set and Gabor-Based Active Contour Algorithm for Segmenting Textured Images," UCLA CAM Report 02-39, 2002. ftp://ftp.math.ucla.edu/pub/camreport/cam02-39.ps.gz

[13] G. Chung, L.A. Vese, "Energy Minimization Based Segmentation and Denoising Using a Multilayer Level Set Approach," Energy Minimization Methods in Computer Vision and Pattern Recognition, vol. 3757/2005, pp. 439–455, 2005. http://dx.doi.org/10.1007/11585978_29

[14] L. He, S. Osher, "Solving the Chan-Vese Model by a Multiphase Level Set Algorithm Based on the Topological Derivative," Proceedings of SSVM'07, vol. 4485/2007, pp. 777–788, 2007. http://dx.doi.org/10.1007/978-3-540-72823-8_67

[15] N.E. Zehiry, S. Xu, P. Sahoo, A. Elmaghraby, "Graph cut optimization for the Mumford-Shah model," Proceedings of VIIP, ACTA Press, pp. 182–187, 2007. http://webdocs.cs.ualberta.ca/~nray1/CMPUT605/track2_papers/MS_GraphCut.pdf

[16] N. Badshah, K. Chen, "Multigrid Method for the Chan-Vese Model in Variational Segmentation," Communications in Computational Physics, vol. 4, no. 2, pp. 294–316, 2008.

[17] Bae, Tai, "Efficient Global Minimization for the Multiphase Chan-Vese Model of Image Segmentation," Energy Minimization Methods in Computer Vision and Pattern Recognition, vol. 5681, pp. 28–41, 2009. http://dx.doi.org/10.1007/978-3-642-03641-5_3

[18] M. Keegan, B. Sandberg, T.F. Chan, "A Logic Framework for Multiphase Multichannel Image Segmentation," UCLA CAM Report 10-60, 2010. ftp://ftp.math.ucla.edu/pub/camreport/cam10-60.pdf