HIGHLY ACCURATE SEGMENTATION USING GEOMETRIC ATTRACTION-DRIVEN FLOW IN EDGE-REGIONS

By

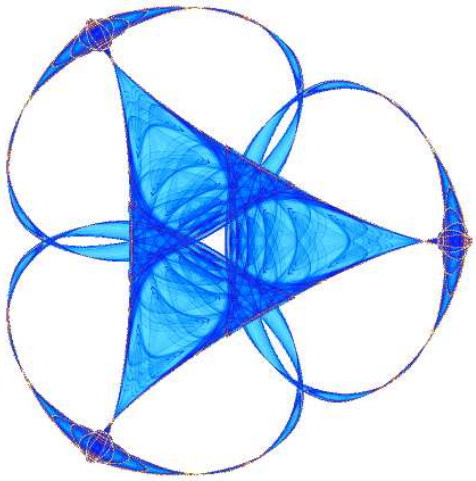
Jooyoung Hahn

and

Chang-Ock Lee

IMA Preprint Series # 2125

(June 2006)



INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS

UNIVERSITY OF MINNESOTA

400 Lind Hall

207 Church Street S.E.

Minneapolis, Minnesota 55455–0436

Phone: 612-624-6066 Fax: 612-626-7370

URL: http://www.ima.umn.edu

Highly Accurate Segmentation Using Geometric Attraction-Driven Flow in Edge-Regions*

Jooyoung Hahn† and Chang-Ock Lee‡

Division of Applied Mathematics
KAIST, Daejeon, 305-701, Korea

Abstract

A highly accurate segmentation algorithm is proposed for extracting objects from an image that has simple background colors or simple object colors. Two main concepts, geometric attraction-driven flow (GADF) and edge-regions are combined to detect exact boundaries of objects in a sub-pixel resolution. GADF gives exact locations of boundaries and edge-regions help to make initial curves close to objects. For high accuracy in segmentation, we additionally propose a local region competition algorithm which detects perceptible boundaries of objects. The whole algorithm is able to extract objects even though there are weak edges, shadows, and highly non-convex shapes. Furthermore, there are no manipulations of parameters in the whole process.

Keywords: GADF, edge-regions, accurate segmentation, local region competition, level set method

1 Introduction

In the segmentation problems to extract objects from an image to make, for examples, 3D VR (virtual reality) contents or to estimate sizes of objects, a key issue is the accuracy which means that we segment an image along the perceptible boundaries of objects. Our research is motivated by making 3D VR contents of commercial products. It makes an e-catalog that customers can browse a product in three dimensional virtual space on internet markets. A common way of making a 3D VR content starts from taking hundreds of photographs of the product with different view angles in a photo studio. The most difficult step is to extract the product from a background with high accuracy. The images taken in the studio have well-known difficulties in segmentation problems. These mainly come from the lighting condition in the studio and the complex shape of the product. Most of lighting conditions make shadows which cause weak boundaries between objects and the background. More serious weak boundaries are produced by a reflection on some parts of an object due to bright lighting conditions and properties of materials of the object. It changes colors of objects into almost white which is normally used as a background color. In

*This work was supported by KRF-2005-213-C00004 and in part by IMA with funds provided by NSF.
†jyhahn76@amath.kaist.ac.kr
‡colee@amath.kaist.ac.kr

addition, there are other difficulties; shapes of objects can be highly non-convex or multiply connected.

There have been a lot of segmentation algorithms. The snake model in [1] is a foundation of curve evolution based on the minimization of an energy. Geodesic active contours was introduced in [2], however, the curves may pass over weak edges because edge functions can not give same values on both weak and strong edges. In [3], the region-aided geometric snake was proposed for a fast convergence in the non-convex boundary and more robust detection of weak edges. The results depend on a region map that an end user should choose. In [4], a curvature vector flow was used to overcome a limitation of [3] for segmenting highly non-convex shapes. However, the method is applicable only to strong edges of shapes.

There is a different approach in [5], which does not use edge functions. The minimizer in the energy functional can be a possible solution of a leakage problem near weak edges and detects non-convex shapes. However, the formulation has four parameters which depend on images. In [6], a statistical approach was proposed, which is similar to [5]. It assumes that an image is represented by the Gaussian mixture model, that is usually not valid for photographs taken in the studio. In [7], geodesic active regions was introduced, which is a clever combination of geodesic active contours in [2] and the statistical approach in [6]. It still has parameters that should be manipulated to make an adequate segmentation of objects.

In this paper, we propose a segmentation algorithm which is highly accurate and does not have any manipulation of parameters. It captures most of boundaries regardless of strength of edges and concavity of boundaries. There are two main concepts, geometric attraction-driven flow (GADF) and edge-regions, which are combined to detect exact boundaries of objects in a sub-pixel resolution. The flow shows the exact location of boundaries of objects. If we consider an image as a two dimensional manifold, we obtain GADF by comparing two lengths of curves along the direction of the largest change in the manifold. Edge-regions contain most of edges. We compute inward fluxes in the gradient field of a strength of edges to obtain such regions. Since the orientations of GADF far from boundaries of objects do not point to the boundaries, we construct initial curves very close to the boundaries using the edge-regions. It naturally solves a problem of a slow convergence in highly non-convex boundaries and a dependence on initial curves. After we obtain the GADF and initial curves, we solve a simple advection equation of curves in the flow to segment objects. The evolving curves move to the exact boundaries which divide objects and background. According to the purpose of segmentation, for examples, accurate extraction of objects or measurement of sizes of objects, we additionally propose a local region competition algorithm to obtain perceptible boundaries of objects. We have successfully accomplished the segmentation of objects from images taken in the studio. Our algorithm can be applied to other kinds of segmentation problems by taking the appropriate strategy for selecting the edge-regions. An example is to extract aphids from images of soybean leaves. We may count the number of aphids that live on the sampled leaves and obtain an exact size of each aphid. With those information, farmers can get the appropriate time to dust powder.

The rest of this paper consists of following sections. We derive GADF in Section 2.1 and detect edge-regions in Section 2.2. We solve a partial differential equation (PDE) in Section 2.3 to obtain curves which connect the edge-regions. The curves will be used as an initial guess for the next step. In Section 2.4, we segment images. According to purposes of

segmentation, a post processing is applied in Section 2.5. Examples for whole process and numerical aspects are illustrated in Section 3. The paper is concluded in Section 4.

2 Algorithms

Our algorithm consists of five steps to extract objects from an image with high accuracy. In particular, each step does not have any manipulation of parameters.

2.1 Step 1: Derivation of GADF

In this section, we will define exact boundaries of objects and derive GADF. First, we consider a scalar image $I\colon \Omega \subset \mathbf{R}^2 \to \mathbf{R}$, which is smooth enough. Since we may assume that the intensity near boundaries of an object changes rapidly from one homogeneous region to another homogeneous region, we define a point x in Ω as an edge point if the second derivative of I at x along $\nabla I(x)$ vanishes.

For a smooth scalar function u on the real line, the following is easily proved: If $u' \neq 0$ and $u'' \neq 0$ in $(x-r, x+r)$ for some $r > 0$, for any positive $\epsilon \leq r$, we have

$$sgn\left(\int_x^{x+\epsilon} u'(t)\,dt - \int_{x-\epsilon}^{x} u'(t)\,dt\right) = sgn\left(u''(x)\right),$$

where

$$sgn(x) \equiv \begin{cases} x/|x| & \text{if} \quad x \neq 0, \\ 0 & \text{if} \quad x = 0. \end{cases}$$

Motivated by this statement, we define the geometric attraction-driven flow (GADF) $\vec{F}(x)$ by

$$\vec{F}(x) = sgn\big(\ell(x)\big)\frac{\nabla I(x)}{|\nabla I(x)|}, \quad \forall x \in \Omega,$$

where

$$\begin{aligned} \ell(x) &= \int_0^1 {c_x}'(s)\,ds - \int_{-1}^0 {c_x}'(s)\,ds, \\ c_x(s) &= I\left(x + s\frac{\nabla I(x)}{|\nabla I(x)|}\right). \end{aligned} \tag{2.1}$$

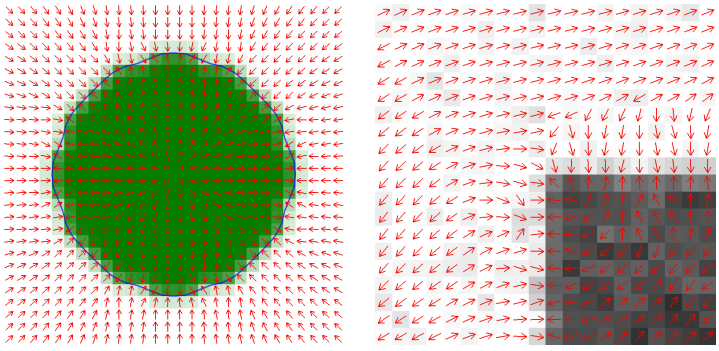
Then GADF runs forward the edge points.

Now, we extend the GADF to a color image $I\colon \Omega \subset \mathbf{R}^2 \to \mathbf{R}^3$. As in the case of a scalar image, we define GADF $\vec{F}(x)$ by

$$\vec{F}(x) = sgn\big(\ell(x)\big)v_\Lambda(x),$$

where

$$\begin{aligned} \ell(x) &= \int_0^1 \left|{c_x}'(s)\right| ds - \int_{-1}^0 \left|{c_x}'(s)\right| ds, \\ c_x(s) &= I\big(x + sv_\Lambda(x)\big). \end{aligned} \tag{2.2}$$



(a) (b)

Figure 2.1: Exact boundary and GADF: The closed curve in (a) represents the exact boundary of an object and GADF runs against along the exact boundary. The vector in (b) is GADF near a corner.

Here, $v_\Lambda(x)$ and $v_\lambda(x)$ are normalized eigenvectors corresponding to the maximum eigenvalue $\Lambda(x)$ and the minimum eigenvalue $\lambda(x)$, respectively, in a diffused tensor $\mathbf{E}$ of the metric tensor $\sum_{k=1}^{3} \nabla I_k \nabla {I_k}^T$ at x in Ω, where T denotes the transpose. Note that two definitions of $\ell(x)$, (2.1) and (2.2) are the same except the dimension of range of I. The diffused tensor $\mathbf{E}$ can be obtained by solving the following isotropic nonlinear diffusion equations in [8]:

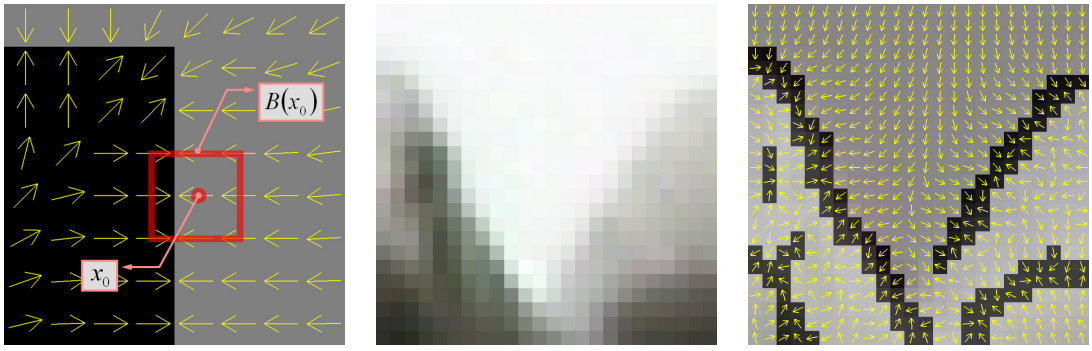
$$\frac{\partial \mathbf{E}(x,t)}{\partial t} = \nabla \cdot \left(g \left(\sum_{i,j=1}^{2} |\nabla E_{ij}(x,t)|^2 \right) \nabla \mathbf{E}(x,t) \right), \tag{2.3}$$

where $g(x) = 1/\sqrt{1+x}$.

We define the exact boundaries of objects in a color image as closed curves near objects to connect the points where GADF runs against each other. See a curve and orientation of GADF in Figure 2.1-(a). The main feature of GADF is the orientation of vectors near to the boundaries. If we regard an image I in a small neighborhood of x as a local coordinate patch for representing two dimensional manifold, the flow $\vec{F}(x)$ is obtained by selecting one of $\pm v_\Lambda(x)$, which gives a longer curve in the manifold among $I(x+sv_\Lambda(x))$ and $I(x-sv_\Lambda(x))$, $0 \leq s \leq 1$. Since changes of colors are large near boundaries of objects, it gives $\vec{F}(x)$ which points to the boundaries. In Figure 2.1-(b), we illustrate an example of the GADF. The vectors show that the flow runs against each other near boundaries of objects.

2.2 Step 2: Detection of edge-regions

Edge-regions are roughly defined as a union of regions that include most of boundaries of objects, such as weak edges and strong edges. There are two steps to detect edge-regions. The first step is to select candidates of edge-regions. The candidates of edge-regions contain points that are unnecessary or even harmful for the segmentation process. These points usually come from small variation of the intensity such as noises. The second step is to delete such bad candidates of edge-regions.



(a) (b) (c)

Figure 2.2: In (a), an ideal case of normalized vectors $\vec{w}/|\vec{w}|$ around x_0 which represents an edge are shown. (b) is a small part of Figure 3.4. The vector in (c) is the normalized vector field $\vec{w}/|\vec{w}|$ and the black region in (c) represents the candidates of edge-regions.

2.2.1 Selection of candidates of edge-regions

In this section, we will introduce an algorithm for selecting candidates of edge-regions. First, we measure two different fluxes of vector fields $\vec{w}$ and $\vec{w}/|\vec{w}|$ on a rectangle $B(x)$:

$$f(x) = \int_{B(x)} \nabla \cdot \vec{w}\, dx \quad \text{and} \quad \bar{f}(x) = \int_{B(x)} \nabla \cdot \frac{\vec{w}}{|\vec{w}|}\, dx,$$

where

$$\vec{w}(x) \equiv \nabla\big(\Lambda(x) + \lambda(x)\big) \tag{2.4}$$

and $B(x)$ is a union of x and eight neighbor pixels around x. Note that both of Λ and λ are small in homogeneous regions, and Λ is much larger than λ in regions near edges. At corners, Λ and λ are both large; see [8].

To see how the candidates of edge-regions are formed, let us first consider a set

$$E_1 \equiv \left\{ x \in \Omega \;\middle|\; \left[\bar{f}(x)\right]^- > C_{\bar{f}} \right\}. \tag{2.5}$$

Here, $[x]^-$ denotes the negative part of x, i.e., $[x]^- = \max(-x, 0)$. The sum of two eigenvalues in (2.4) represents the strength of edges in a color image. The magnitude of gradient of the sum is the maximum rate of change in the strength. Magnitude of the vector $\vec{w}$ depends on weak edges and strong edges. However, the normalized vector $\vec{w}/|\vec{w}|$ clearly gets rid of the dependence on the strength of edges. By using calculating the inward flux in the normalized vector field, we obtain the same measurement for weak edges and strong edges. If a point is close to edges, vectors around the point make an inward flux. The negative part of $\bar{f}$ gives the strength of inward flux. A large inward flux $\left[\bar{f}(x)\right]^-$ makes a point x included in E_1.

Now, we explain how to choose the constant value $C_{\bar{f}}$ in (2.5). If we compute $\bar{f}(x)$ for all of possible normalized vectors around x, the probability density function of $\{\bar{f}(x)\}$ becomes a Gaussian distribution with a zero mean and a deviation $\sigma \approx 1.565$. Considering an ideal

case of a normalized vector field which represents an edge in Figure 2.2-(a), the value of $\left[\bar{f}(x_0)\right]^-$ is $2+2\sqrt{2}$. It is almost three times the deviation σ, which means a chance for the ideal case is extremely rare. It shows that $C_{\bar{f}}$ should be chosen to be reasonably smaller than 3σ. We choose $C_{\bar{f}} = 1.565$. It means that if eight vectors around x make $\left[\bar{f}(x)\right]^-$ greater than $C_{\bar{f}}$, a chance for such eight vectors is about 16 percent among all possible cases. Note that the value $C_{\bar{f}}$ was chosen regardless of the strength of edges.

In spite of merit to use the normalized vectors $\vec{w}/|\vec{w}|$ to form E_1, the set E_1 may contain some points where the image changes slightly. Since the magnitude of $\vec{w}$ represents strength of edges, we define a set

$$E_2 \equiv \left\{x \in \Omega \,|\, |f(x)| \le C_f\right\},$$

where C_f is a positive constant determined in the following. We want to decide the constant C_f as an upper bound of value $|f(x)|$ for indicating regions where the image changes slightly. By using a simple estimation in discrete sense, we get

$$|f(x)| \le 16 \max_{y \in B(x)} \sum_{k=1}^{3} |\nabla I_k(y)|^2, \quad \forall x \in \Omega.$$

If we admit heuristically that $|\nabla I_k| \le 1$ for all k's in the regions where the image changes slightly, we obtain the constant $C_f = 48$.

Now, we define the candidates of edge regions

$$E_c \equiv E_1 \setminus E_2 = \left\{ x \in \Omega \,\middle|\, \min\left(\left[\bar{f}(x)\right]^-, \frac{C_{\bar{f}}}{C_f}\left[f(x)\right]^- \right) > C_{\bar{f}} \right\}.$$

If we define

$$S(x) \equiv \exp\left(-\min\left(\left[\bar{f}(x)\right]^-, \frac{C_{\bar{f}}}{C_f}\left[f(x)\right]^- \right)\right), \tag{2.6}$$

$$\delta \equiv \exp\left(-C_{\bar{f}}\right) \simeq 0.209, \tag{2.7}$$

then

$$E_c = \left\{x \in \Omega \,|\, S(x) < \delta\right\}. \tag{2.8}$$

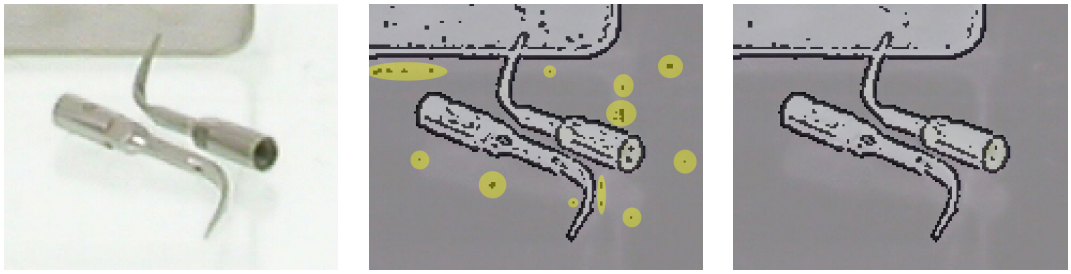
Note that

$$S(x) \approx \begin{cases} 0 & \text{near edges,} \\ 1 & \text{far from edges.} \end{cases}$$

We illustrate the normalized vector field $\vec{w}/|\vec{w}|$ and candidates of edge-regions in Figure 2.2-(c).

2.2.2 Deletion of bad candidates

In this section, we will delete bad candidates in (2.8) and obtain the formal definition of edge-regions. Bad candidates are roughly considered as points in E_c, which are not regarded



(a) (b) (c)

Figure 2.3: Bad candidates and edge-regions: (a) is an original image. The black region in (b) represents candidates of edge-regions and the candidates in yellow areas are bad candidates which are not parts of boundaries of objects obviously. The black regions in (c) are the edge-regions.

as edges such as yellow areas Figure 2.3-(b). In order to delete the bad candidates, we propose an algorithm by observing features of bad candidates.

Bad candidates are characterized by two features. We divide E_c into disjoint connected components

$$E_c = \bigcup_{i=1}^{K} \Omega_i. \tag{2.9}$$

The first feature is that bad candidates are hardly included in a component which has a large number of pixels in discrete sense. Since the normalized vector field $\vec{w}/|\vec{w}|$ can make large inward flux continuously along strong edges, the component which has a large number of pixels should be a long part of boundary of an object. The second is that the image in a small neighborhood of bad candidates does not change significantly. We use these features to construct sets deleted from Ω_i's.

For the rest of the paper, we denote the average and the deviation of a set A by

$$\begin{aligned} \mu(A) &\equiv \text{ the average of } A, \\ \sigma(A) &\equiv \text{ the deviation of } A, \end{aligned}$$

respectively. We compute the number of pixels and the deviation of gray values of each component in (2.9):

$$\begin{aligned} N_i &\equiv \text{the number of pixels in } \Omega_i, \\ \Sigma_i &\equiv \sigma\big(\{I_g(x) \mid x \in \Omega_i\}\big), \end{aligned}$$

where $I_g(x)$ is a gray value of a color image at x. Two values, μ_N and σ_N are the average and the deviation of numbers of pixels for all components, respectively. Two values, μ_Σ and σ_Σ are the average and the deviation of deviations of gray values for all components, respectively. That is,

$$\begin{aligned} \mu_N \equiv \mu\big(\{N_i \mid i = 1, ..., K\}\big) \text{ and } \sigma_N \equiv \sigma\big(\{N_i \mid i = 1, ..., K\}\big), \\ \mu_\Sigma \equiv \mu\big(\{\Sigma_i \mid i = 1, ..., K\}\big) \text{ and } \sigma_\Sigma \equiv \sigma\big(\{\Sigma_i \mid i = 1, ..., K\}\big). \end{aligned}$$

Now, we introduce the algorithm that constructs sets of points deleted from E_c. For a point x in a component Ω_i, we compute the deviation of gray values on locally sampled pixels. That is,

$$\Sigma^{loc}(x) \equiv \sigma\big(\{I_g(y) \mid y \in B(x)\}\big).$$

We define a set D_i which will be deleted from Ω_i by

$$D_i \equiv \left\{ x \in \Omega_i \,\middle|\, \frac{S(x)}{N_i} > \frac{\delta}{n_i(x)} \right\},$$

where δ is the value in (2.6). A threshold $n_i(x)$ is defined as follow:

$$\begin{aligned} n_i(x) &\equiv \mu_N + 2\alpha\left(-\frac{\Sigma^{loc}(x) - \mu_\Sigma}{\sigma_\Sigma}\right)\sigma_N, \\ \alpha(x) &\equiv \frac{1}{1+e^{-x}}. \end{aligned} \tag{2.10}$$

Note that the threshold depends on the deviation of gray values around each point and prevents points in a small component including edges from being deleted.

Finally, we obtain a formal definition of edge-regions:

$$\mathcal{R} \equiv \bigcup_{i=1}^{K} \mathcal{R}_i, \quad \mathcal{R}_i \equiv \Omega_i \setminus D_i. \tag{2.11}$$

Candidates of edge-regions and edge-regions are illustrated in Figure 2.3.

2.3 Step 3: Connection of edge-regions

The main goal in this section is to obtain regions which include all boundaries of objects by connecting edge-regions in (2.11) and obtain a good initial curve for the segmentation process in the next section. In [9], line connection algorithms for contour completion were proposed by using an anisotropic diffusion operator. With the algorithms, the edge-regions become thick due to the diffusion process and it is hard to decide how large areas to be regarded as new edge-regions. Instead, we propose a Hamilton-Jacobi equation to obtain connected regions from edge-regions.

We denote the zero level set of a continuous function $\psi\colon \Omega \to \mathbf{R}$ by Γ_ψ. We introduce a Hamilton-Jacobi equation:

$$\frac{\partial}{\partial t}\psi(x,t) + F_s(x)F_l(x,t)\big|\nabla\psi(x,t)\big| = 0 \quad \text{in} \quad \Omega \times (0, T_1]. \tag{2.12}$$

The initial condition is given by a signed distance function which is zero on a boundary of the edge-regions $\mathcal{R}$, negative inside $\mathcal{R}$, and positive outside $\mathcal{R}$. In Figure 2.4, we illustrate an example to present the curves $\Gamma_{\psi(\cdot,0)}$ with edge-regions, a result $\Gamma_{\psi(\cdot,T_1)}$ of (2.12), and an initial curve $\Gamma_{\widetilde{\psi}}$ for the segmentation process.

The time dependent local force F_l is defined by

$$F_l(x,t) \equiv \left(G_\sigma * H_\epsilon\left(\big|v_\lambda(x)\cdot\nabla\psi(x,t)\big| - \frac{\sqrt{2}}{2}\right)\right) H_\epsilon\left(\kappa_\psi(x,t) - \frac{1}{2}\right), \tag{2.13}$$

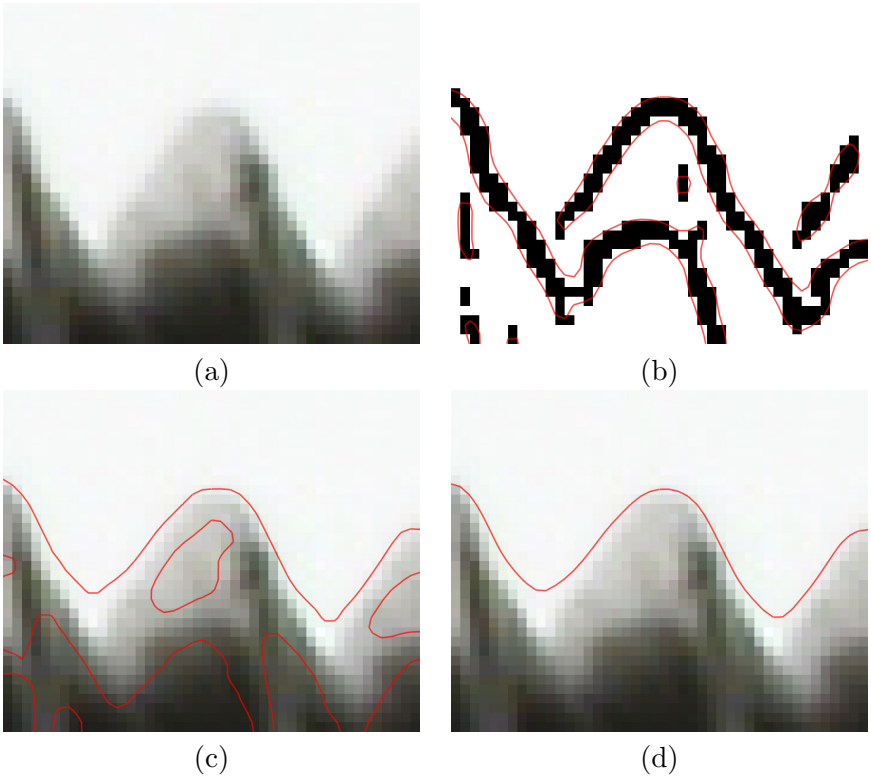
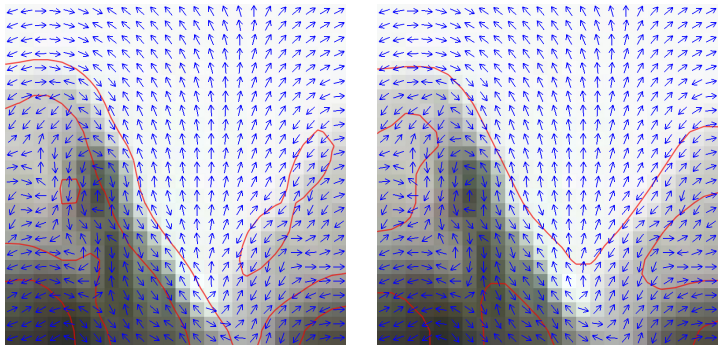


Figure 2.4: Procedures of solving (2.12): (a) is a small part of Figure 3.4. The black regions in (b) are edge-regions and the curves $\Gamma_{\psi(\cdot,0)}$ in (b) are the initial condition for (2.12). The curves $\Gamma_{\psi(\cdot,T_2)}$ in (c) are a result of (2.12), which connects edge-regions along boundaries of objects. The curve $\Gamma_{\widetilde{\psi}}$ in (d) is obtained from the curves in (c) and will be used as an initial curve for the segmentation process.



(a) (b)

Figure 2.5: Eigenvectors v_λ and evolving curves $\psi(x,t)$: The concave parts of curves will stay because of the restriction in (2.14). The curves in (a) are initial condition for (2.12) and the curves in (b) are obtained at time T_1. Note that the image is a small part of Figure 3.4.

where $G_\sigma * f$ is the convolution of a function f with the two-dimensional Gaussian kernel with $\sigma = 1$. Note that $v_\lambda(x)$ is the eigenvector corresponding to the minimum eigenvalue of the diffused tensor $\mathbf{E}$ and $\kappa_\psi(x,t)$ is the curvature of level curves of $\psi(x,t)$. The function H_ϵ is a smeared-out Heaviside function:

$$H_\epsilon(x) \equiv \frac{1}{2}\left(1 + \frac{2}{\pi}\tan^{-1}\left(\frac{x}{\epsilon}\right)\right),$$

where $\epsilon = 0.001$, which is the size of the bandwidth of numerical smearing.

Now, we explain how (2.12) makes curves connect edge-regions along boundaries of objects. In (2.13), the first term

$$H_\epsilon\left(\left|v_\lambda(x)\cdot\nabla\psi(x,t)\right| - \frac{\sqrt{2}}{2}\right)$$

forces curves to move forward along boundaries of objects since v_λ represents a parallel direction to the isophote line of a color image. The force exerts on points where the angle between $\nabla\psi$ and v_λ is less than $\pi/2$. Unfortunately, concave parts of curves may also move to far from boundaries. See the direction of v_λ near the concave parts of objects in Figure 2.5. Therefore, we prohibit the movement along boundaries to concave parts of curves by using the second term in (2.13):

$$H_\epsilon\left(\kappa_\psi(x,t) - \frac{1}{2}\right). \tag{2.14}$$

The force F_s based on statistics is defined by

$$F_s(x) \equiv \alpha\left(\frac{\Sigma^{loc}(x) - \mu_\Sigma}{\sigma_\Sigma}\right),$$

where $\alpha(x)$ is the function in (2.10). We show an example in Figure 2.6 to explain a role of the force F_s. Since the edge-regions may capture a part of shadow, there can be initial

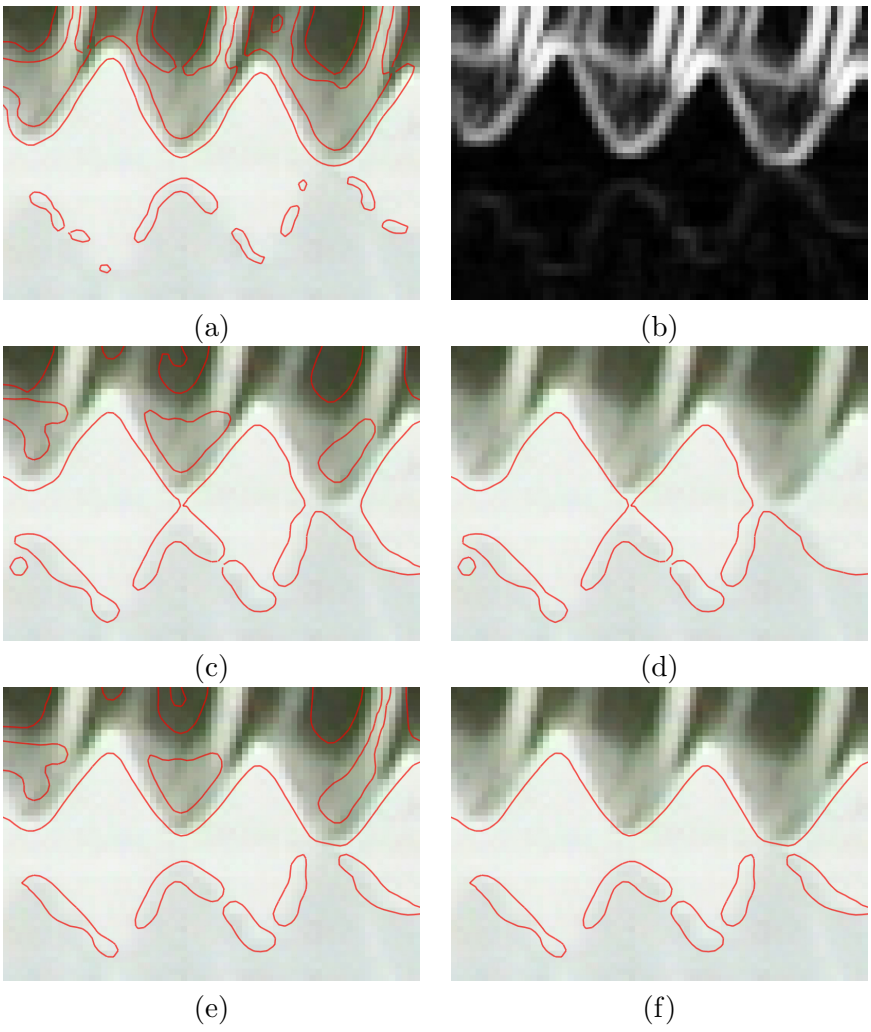


Figure 2.6: The effect of the force $F_s(x)$ on the result of (2.12): a result without using $F_s(x)$ has excessive connections. The curves in (a) are the initial condition $\Gamma_{\psi(\cdot,0)}$ for (2.12). The profile in (b) is $F_s(x)$. The curves in (c) are a result in (2.12) without using $F_s(x)$. The curve in (d) from (c) is a bad initial curve for the segmentation process. The curves in (e) are a result in (2.12) with $F_s(x)$. The curves in (f) from (e) are good initial curves for the segmentation. Note that the image is a small part of Figure 3.4.

curves $\Gamma_{\psi(\cdot,0)}$ placed outside of objects. If we use only the force F_l, the isolated regions in the middle of (a) are connected to the boundaries of objects. Then, we obtain outer contours of regions, which are not close to the boundaries of objects. In order to restrict unnecessary movement of curves, we use exactly the same idea as we use a threshold $n_i(x)$ in (2.10). If $\Sigma^{loc}(x)$ is much smaller than μ_Σ, changes of colors near x are small. It means that the neighborhood of x cannot be considered as a part of edges. Therefore $F_s(x)$ is close to 0. On the other hand, if $\Sigma^{loc}(x)$ is much larger than μ_Σ, $F_s(x)$ has a value close to 1. The force F_s is adaptive to images because values of μ_Σ and σ_Σ are an appropriate statistical representations of edges for each image. In Figure 2.6, we show a profile of $F_s(x)$ on the domain of image, a difference between the result in (2.12), and a result without using $F_s(x)$.

Remark: the motion of curves also may be restricted where surface of an object is reflected. If edge-regions in those parts are not connected, we locally apply the same algorithm to connect them.

2.4 Step 4: Segmentation using GADF in edge-regions

For the regions that we obtained in the previous section, we take outer contours of each connected region except those nested by other regions. Additionally, we take inner contours of connected regions nested by other regions if the nested one contains edges of holes in objects. We denote the union of such contours by $\Gamma_{\widetilde{\psi}}$. In this section, we will solve a simple advection equation in order to segment objects:

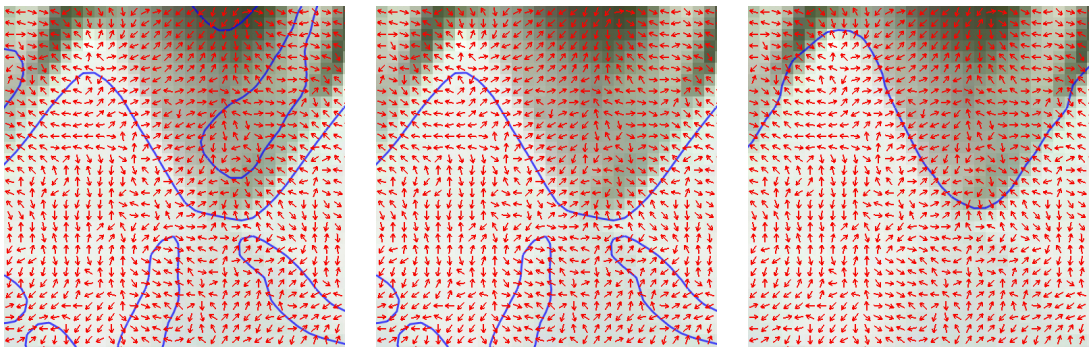
$$\begin{aligned} &\frac{\partial}{\partial t}\phi(x,t) + \vec{F}(x)\cdot\nabla\phi(x,t) = 0 \quad \text{in} \quad \Omega\times(0,T_2], \\ &\phi(x,0) = \widetilde{\psi}(x) \quad \text{in} \quad \Omega, \end{aligned} \tag{2.15}$$

where $\vec{F}$ is the GADF and $\widetilde{\psi}(x)$ is a signed distance function which has the zero level set as $\Gamma_{\widetilde{\psi}}$.

If a curve in $\Gamma_{\widetilde{\psi}}$ is a contour of a simply connected region, it shrinks and disappears by (2.15) since the simply connected region wraps the flow $\vec{F}(x)$ which runs against each other around edges of the image. On the other hand, if a curve in $\Gamma_{\widetilde{\psi}}$ is a contour of a multiply connected region, it moves to the exact boundaries of objects and stays there since the multiply connected region contains the flow $\vec{F}(x)$ which runs against each other along the exact boundaries of objects; see Figure 2.7.

2.5 Step 5: Post processing

In the previous section, we may consider the result in Figure 2.7-(c) as a proper segmentation. However, the result is not good enough for extracting objects from images because high accuracy of segmentation is required to make, for examples, 3D VR contents or to estimate sizes of objects. In the previous section we found the exact boundaries of objects, which are inflection points along ∇I in the case of gray image. However, people usually recognize a little bit outside as borders of objects rather than the exact boundaries because human vision perceives objects without missing any part of the objects. We call such borders as perceptible boundaries of objects; see Figure 2.8. The significant difference



(a) (b) (c)

Figure 2.7: Initial curves and the result of (2.15): While the outer contours of simply connected regions are disappeared, those of multiply connected regions stay at exact boundaries of objects. The vectors are GADF. A result $\Gamma_{\psi(\cdot,T_1)}$ from (2.12) is in (a). The curves in (b) are initial curve $\Gamma_{\phi(\cdot,0)}$ for (2.15). The curve in (c) are a result of segmentation from (2.15). Note that the image is a small part of Figure 2.6.

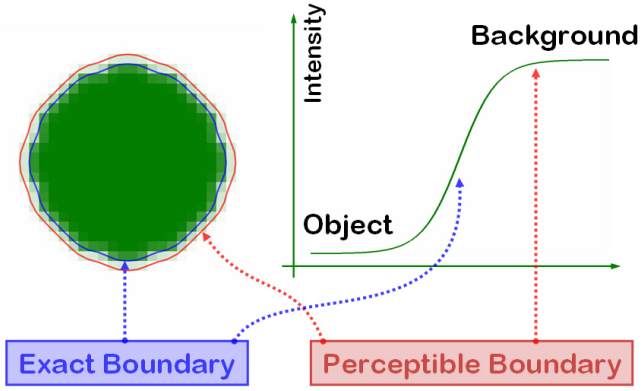


Figure 2.8: The significant difference of the exact boundaries and the perceptible boundaries.

of the exact boundaries and the perceptible boundaries happens when colors of boundaries of objects are changed gradually near edges or under bright lighting conditions in a photo studio.

In this section, we will obtain perceptible boundaries of objects by using a local region competition algorithm based on comparison of local probability density functions; see [10]. The local region competition algorithm gives a PDE which is the same type as (2.12). The proposed PDE is

$$\begin{aligned} &\frac{\partial}{\partial t}\varphi(x,t) = \hat{F}_s(x,t)\big|\nabla\varphi(x,t)\big| \quad \text{in} \quad \Omega\times(0,T_3],\\ &\varphi(x,0) = \frac{1}{4}\phi(x,T_2) + \frac{3}{4}\widetilde{\psi}(x) \quad \text{in} \quad \Omega, \end{aligned} \tag{2.16}$$

where $\phi(x,T_2)$ is the result of (2.15) and $\widetilde{\psi}$ is the initial condition in (2.15). Note that the initial curve $\Gamma_{\varphi(\cdot,0)}$ is chosen closer to $\Gamma_{\widetilde{\psi}}$ rather than $\Gamma_{\phi(\cdot,T_2)}$. We denote inside and outside regions near $\Gamma_{\varphi(\cdot,t)}$ by

$$\begin{aligned} \Omega_\gamma^+(t) &\equiv \big\{x \mid 0 \le \varphi(x,t) \le \gamma\big\},\\ \Omega_\gamma^-(t) &\equiv \big\{x \mid -\gamma \le \varphi(x,t) \le 0\big\}. \end{aligned}$$

We compute the average and the deviation of locally sampled gray values for all $x \in \Omega_\gamma^+(t)\cup\Omega_\gamma^-(t)$:

$$\begin{aligned} \mu_\pm(x,t) &\equiv \mu\big(\{I_g(y) \mid y \in B_\beta(x)\cap\Omega_\gamma^\pm(t)\}\big),\\ \sigma_\pm(x,t) &\equiv \sigma\big(\{I_g(y) \mid y \in B_\beta(x)\cap\Omega_\gamma^\pm(t)\}\big), \end{aligned} \tag{2.17}$$

where $B_\beta(x)$ is defined by a disk at a center x with a radius β. The values of β and γ are empirically chosen as 6 and 3, respectively.

Now, we define the force $\hat{F}_s(x,t)$ in (2.16) by

$$\hat{F}_s(x,t) \equiv G_\sigma * \hat{f}(x,t)$$

where

$$\hat{f}(x,t) \equiv \begin{cases} s_\epsilon\left(\nabla\cdot\left(\frac{\nabla\varphi(x,t)}{|\nabla\varphi(x,t)|}\right)\right) & \text{if} \quad |\,\mu_+ - \mu_-\,| < \varepsilon,\\ s_\epsilon\big(\mathcal{P}_{x,t}^+(x) - \mathcal{P}_{x,t}^-(x)\big) & \text{otherwise.} \end{cases}$$

The function $s_\epsilon(x)$ is a smeared-out sign function defined by $2H_\epsilon(x)-1$. The functions $\mathcal{P}_{x,t}^\pm(y)$ are defined by Gaussian probability density functions:

$$\mathcal{P}_{x,t}^\pm(y) \equiv \frac{1}{\sqrt{2\pi}\sigma_\pm(x,t)}\exp\left(-\frac{(I_g(y)-\mu_\pm(x,t))^2}{2\sigma_\pm(x,t)^2}\right) \tag{2.18}$$

which represent local distributions of sampled gray values around the point x outside and inside of curves $\Gamma_{\varphi(\cdot,t)}$, respectively. If the averages μ_+ and μ_- are too close, we make the curve straight instead of using the difference of two Gaussian probability density functions. We properly choose the value $\varepsilon = 3$.

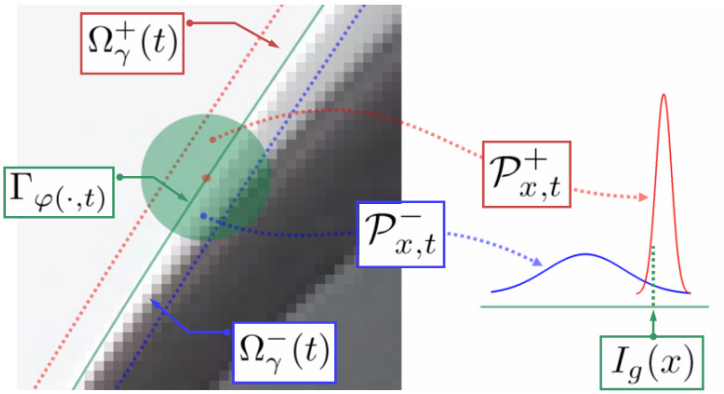


Figure 2.9: Conceptual diagram for understanding the force $\hat{F}_s$ in (2.16). The curve $\Gamma_{\varphi(\cdot,t)}$ is going to move inward.

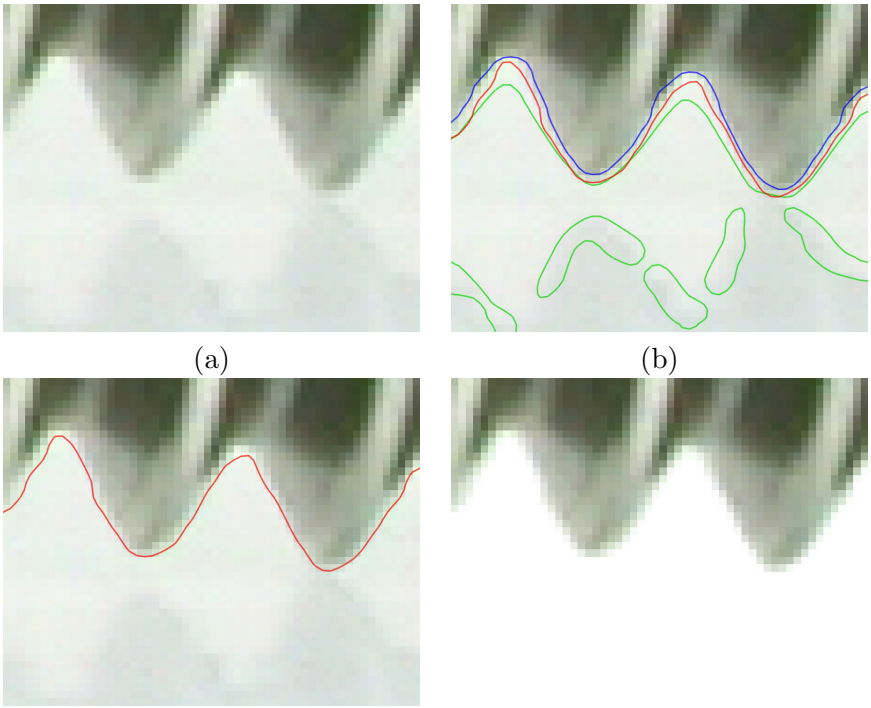


Figure 2.10: The perceptible boundaries of objects and the extracted object: The image in (a) are a small part of Figure 3.4. In two images (b) and (c), the green curves are $\Gamma_{\widetilde{\psi}}$, the blue curve is $\Gamma_{\phi(\cdot,T_2)}$ in (2.15), and the red curve $\Gamma_{\varphi(\cdot,T_3)}$ is the result of (2.16). The extracted object is on white background in (d).

The force $\hat{F}_s$ basically forces each point on curves $\Gamma_{\varphi(\cdot,t)}$ to move inward or outward based on the difference of two local Gaussian probability density functions. If a gray value at a point x in Ω_γ is close to the average $\mu_+(x,t)$, then the point has a high probability of being placed into outside of curves $\Gamma_{\varphi(\cdot,t)}$. In this case, it is natural to force curves $\Gamma_{\varphi(\cdot,t)}$ to move inward. That is,

$$\mathcal{P}^+_{x,t}(x) > \mathcal{P}^-_{x,t}(x) \Rightarrow \hat{F}_s(x,t) \approx 1.$$

See Figure 2.9. In the other way, if a gray value at a point x in Ω_γ is close to the average $\mu_-(x,t)$, then the curves $\Gamma_{\varphi(\cdot,t)}$ will move outward along normal directions to the curves. In Figure 2.10, we present curves from the result in Step 3 to the result in this section. Extracted objects are also presented. We call the final curves $\Gamma_{\varphi(\cdot,T_3)}$ as the perceptible boundaries of objects.

3 Examples and numerical aspects

We illustrate a whole procedure of the proposed algorithm in Figure 3.1. It has multiply connected regions on top right of object and weak edges due to the strong shadow. The image (a) is the original image. Edge-regions are shown in (b) where original image is overlaid with edge-regions. The curves in (c) shows the outer contours of connected edge regions after Step 3. The curves in (d) is the initial curves for the segmentation process in Step 4. From Step 3 to Step 5, we solve three different PDEs to obtain final curves in (e). We use an explicit Euler scheme for time discretization. For space discretization a simple upwind scheme is used in Step 3 and a nonoscillatory upwind scheme is used in Step 4 and 5; see [11] for details of numerical schemes. We also use the reinitialization equation in [12] only for regions where the force is not zero. It makes a separate segmentation between very close two objects difficult. If we solve the reinitialization equation for the whole image domain, a curve may be merged to adjacent other curves because of numerical dissipation. A stopping criterion is given by measuring the relative L^1 error in a small band [12]:

$$\begin{aligned} E_{\psi^{n+1}} &\equiv \frac{\displaystyle\int_{\Omega^{\psi^n}_\alpha} \left|\psi^{n+1}(x) - \psi^n(x)\right| dx}{\displaystyle\int_{\Omega^{\psi^n}_\alpha} \left|\psi^n(x)\right| dx}, \\ \Omega^{\psi^n}_\alpha &\equiv \Big\{ x \in \Omega \,\Big|\, \left|\psi^n(x)\right| \le \alpha \Big\}, \end{aligned}$$

where $\alpha = 1.5$ and n is an index for time discretization. We practically use both a given upper bound for iterations and the stopping criterion E_ψ. More precisely, in Step 3 and 5, we have either $E_{\psi^n} < 10^{-4}$ or $n > 500$. In Step 4, we use the stopping criterion: $E_{\phi^n} < 10^{-5}$ or $n > 500$. Furthermore, in practice, we use $\sigma_\pm + \varepsilon$ instead of $\sigma_\pm$ in (2.18), where $\varepsilon = 10^{-6}$. The image in (f) is an extracted object on white background.

In Figures 3.2, 3.3, and 3.4, original images and extracted objects are presented. Each image has well-known difficulties in segmentation. In Figure 3.2, the yellow part of object is highly non-convex. Since the initial curves in (2.15) are already located near the object, the proposed algorithm can easily segment highly non-convex shape. In Figure 3.3, even though we use the smooth lighting condition, the original color of shiny surface is changed

Figure 3.1: A procedure of highly accurate segmentation algorithm using GADF in edge-regions. The size of image is 940 by 544.

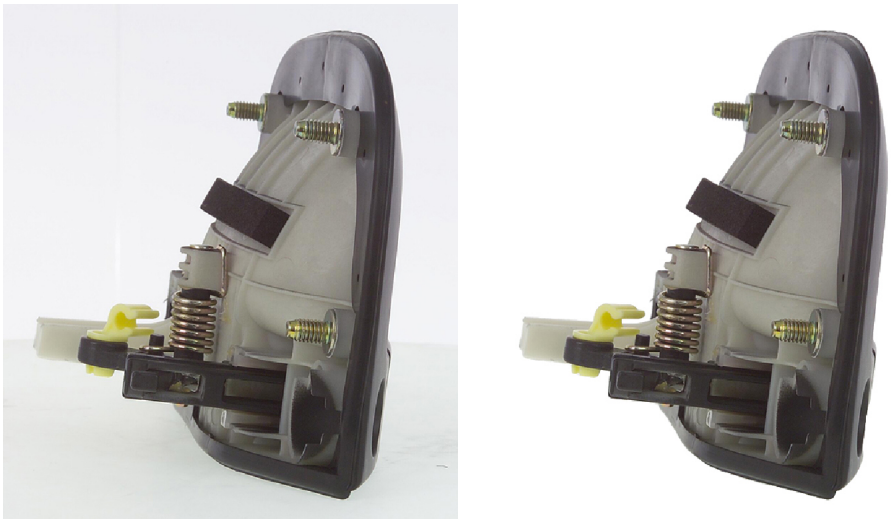


Figure 3.2: An example for 3D VR content: The yellow part is highly non-convex. The size of image is 508 by 576.

Figure 3.3: An example for 3D VR content: Even though we use smooth lighting condition, the original color of shiny surface is changed because of total reflection. It generates weak edges near boundaries of the object. The size of image is 288 by 288.

Figure 3.4: An example for 3D VR content: It has a repeated non-convex shape and weak edges due to the reflection of light. The size of image is 1084 by 488.

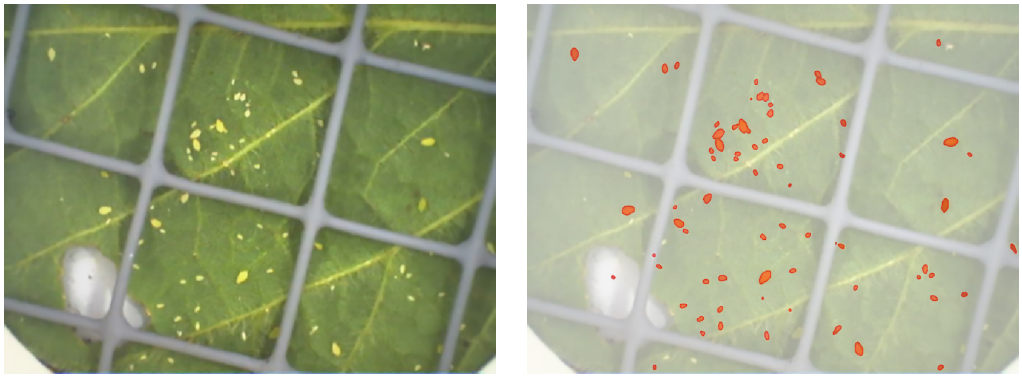


Figure 3.5: Segmenting aphids in a soybean leaf: Original image and segmentation of aphids. The size of each image is 640 by 480.

because of total reflection. It generates weak edges near boundaries of objects. The pot also has strong edges on the handle. Edge-regions can capture both strong and weak edges regardless of the strength of edges. In Figure 3.4, it has a repeated non-convex shape and weak edges due to a reflection of light on a screw thread.

Another kind of an example is illustrated in Figure 3.5, which segments aphids on a soybean leave. We add an additional deletion step at the end of Step 2. Since edge-regions include most of edges, they contain a structure of wire net and veins in Figure 3.5, which are unnecessary parts for segmenting aphids. Since connected components in edge-regions around the wire net are much larger than those around aphids, the components near the wire net can be deleted by selecting connected components whose numbers of pixels are greater than $\mu_N + \sigma_N$. To delete components near veins in edge-regions, we use a ratio of eigenvalues in two-dimensional moment of inertia tensor at the center of mass for each connected component in edge-regions. Since veins have longer and thinner shape than aphids, the components near veins can be deleted by selecting connected components whose maximum eigenvalue in the moment tensor is larger than 36 times the minimum eigenvalue. After we obtain proper edge-regions, we apply the same algorithm in Step 3, 4, and 5 to segment aphids in Figure 3.5.

4 Conclusions

In this paper, we introduced a highly accurate segmentation algorithm for extracting objects from an image that has simple background colors or simple object colors. GADF is a flow obtained by a geometric analysis of eigenspace in a diffused tensor field on a color image as a two-dimensional manifold. It gives the exact locations of boundaries. Edge-regions are obtained by a statistical analysis and include most of edges. GADF and edge-regions are combined to detect exact boundaries of objects in a sub-pixel resolution. For high accuracy in segmentation, we propose a local region competition algorithm which detects perceptible boundaries of objects. The proposed whole algorithm consists of five steps. In the first step, we compute GADF from a diffused tensor field. Edge-regions are computed in the second step. In the third step, we connect edge-regions in order to find an initial curve

close to objects for segmentation. From the initial curve, we obtain the exact boundaries of objects in the forth step. Based on results in Step 3 and 4, we finally obtain the perceptible boundaries of objects in the last step. The proposed whole algorithms is able to extract objects even though there are weak edges, shadows, and highly non-convex shapes. There are no manipulations of parameters in the whole process.

Acknowledgments

The authors would like to thank technical staffs of INTVIM for providing images taken in their studio and Mr. Martin du Saire of USDA-ARS, St. Paul, MN for providing the soybean image.

References

[1] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. J. Comput. Vis.*, 1:321–331, 1988.

[2] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *Int. J. Comput. Vis.*, 22:61–79, 1997.

[3] C. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Trans. Image Processing*, 70:359–369, 1998.

[4] D. Gil and P. Radeva. Curvature vector flow to assure convergent deformable models for shape modelling. In *Lecture Notes in Computer Science, Springer Verlag, Proceedings of EMMCVPR, Lisbon, Portugal 2003*, 2003.

[5] T. Chan and L. Vese. Active contours without edges. *IEEE Trans. Image Processing*, 10:266–277, 2001.

[6] M. Rousson and R. Deriche. A variational frame work for active and adaptive segmentation of vector valued images. Technical report, Research Report, N. 4515, INRIA, July 2002.

[7] N. Paragios and R. Deriche. Geodesic active regions and level set methods for supervised texture segmentation. *Int. J. Comput. Vis.*, 46:223–247, 2002.

[8] T. Brox, J. Weickert, B. Burgeth, and P. Mrázek. Nonlinear structure tensors. *Image and Vision Computing*, 24:41–55, 2006.

[9] D. Gil and P. Radeva. Extending anisotropic operators to recover smooth shapes. *Computer Vis. and Image Understading*, 99:110–125, 2005.

[10] S. C. Zhu and A. Yuille. Region competition: unifying snakes, regions growing, and Bayes/MDL for mutiband image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 18:884–900, 1996.

[11] G. Aubert and P. Kornprobst. *Upwind differencing schemes for hyperbolic conservation laws with source terms.* Springer-Verlag, New York, 2002.

[12] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, 114:146–159, 1994.