Advection-Enhanced Gradient Vector Flow for Active-Contour Image Segmentation

Po-Wen Hsieh[1,*], Pei-Chiang Shao[2] and Suh-Yuh Yang[2,3]

[1] *Department of Applied Mathematics, National Chung Hsing University, South District, Taichung City 40227, Taiwan.*
[2] *Department of Mathematics, National Central University, Jhongli District, Taoyuan City 32001, Taiwan.*
[3] *National Center for Theoretical Sciences, National Taiwan University, Da'an District, Taipei City 10617, Taiwan.*

Abstract. In this paper, we propose a new gradient vector flow model with advection enhancement, called advection-enhanced gradient vector flow, for calculating the external force employed in the active-contour image segmentation. The proposed model is mainly inspired by the functional derivative of an adaptive total variation regularizer whose minimizer is expected to be able to effectively preserve the desired object boundary. More specifically, by incorporating an additional advection term into the usual gradient vector flow model, the resulting external force can much better help the active contour to recover missing edges, to converge to a narrow and deep concavity, and to preserve weak edges. Numerical experiments are performed to demonstrate the high performance of the newly proposed model.

1 Introduction

The technique of snake (also called active contour) was first introduced by Kass *et al.* [14] for image segmentation and it has been widely used in many applications of computer vision and image processing, such as edge detection, shape recognition, and object tracking etc., see e.g., [11, 14, 17, 21, 28]. Roughly speaking, a snake is represented by a contour that can deform to achieve boundary extraction of the desired object through an energy-minimization process. The contour-minimizing energy basically consists of two parts,

---

*Corresponding author. *Email addresses:* `pwhsieh@nchu.edu.tw` (P.-W. Hsieh), `shaopj823@gmail.com` (P.-C. Shao), `syyang@math.ncu.edu.tw` (S.-Y. Yang)

in which the internal energy part, defined by the snake itself, is designed to control the smoothness and tightness of the snake, while the external energy part, represented by a certain image quantity, is used to drive the snake to approach the desired features in the image such as object edges. By using the calculus of variations, a minimizer of the energy functional satisfies a force balance equation which is also consisting of two parts, the internal force and the external force. When the internal and external forces achieve balance, the energy achieves a minimum value.

According to the ways of implementing the snake, there are basically two types of active contours. One is the parametric active-contour [9, 10, 14, 18, 25, 26], where contours are explicitly represented as parameterized curves, and the other is the geometric active-contour [4–6, 20], which is largely inspired by level set methods and represents curves implicitly. In general, the former has much lower computational complexity, so its implementation is efficient in convergence speed. Nevertheless, the latter is capable of handling changes in the topology of contours so that the snake can split or merge naturally during the evolution process for dealing with images with multiple objects.

In this paper, we will focus on the parametric active-contours for image segmentation. Although the parametric snake is able to perform flexible deformations, some major difficulties such as small capture range, deep and narrow concavity convergence and weak-edge leakage still remain and need to be addressed further [8,15,18,19,22,30,34]. To overcome these limitations, the external force field, which is usually written as the gradient of a potential function, plays an important role and it has been extensively improved over the years. In 1998, Xu and Prince [32] proposed the gradient vector flow (GVF), which diffuses the gradient vector field (i.e., external force field) to enlarge the capture range and improves the convergence for entering concavities. However, the snake using GVF external force is not able to go into long and thin indentations (LTIs for short) [23,33] and may cause leakage problems [32]. Due to this reason, Xu and Prince then developed a generalized version of the GVF snake, called the generalized GVF (GGVF) [33], by adding two spatially varying weighting parameters to improve the convergence into LTIs and protect weak edges. Afterwards, Ning *et al.* [23] introduced an improved external force field for snakes, called the GVF in normal direction (NGVF for short). By decomposing the Laplacian operator and adopting only the normal component, the NGVF snake can achieve a faster convergence speed towards concavities, but the convergence capability for LTIs has not been significantly improved. On the other hand, Wang *et al.* [29] studied another new external force field, called the normally biased GVF (NBGVF). It mainly adopts the tangential component of diffusion to preserve weak edges, and then adds the normal component as a biasing weight to maintain some desirable properties of GVF and NGVF. Recently, to take care of both weak-edge preserving and LTI convergence, Wu *et al.* [30] investigated an adaptive diffusion flow (ADF) field by adaptively adopting a harmonic hypersurface minimal functional and an infinity Laplacian functional. The ADF snake possesses several good properties including weak-edge preserving, concavity convergence, and noise robustness. In summary, all the above methods mainly employ diffusion to act on the gradient vector field, in the direction of tangential, normal, or both.

In this paper, borrowing an idea from image denoising, we will present a new type of external force employed in the active-contour image segmentation that applies advection [13,16] to enhance the gradient vectors. To the best of our knowledge, the advection effect has not been investigated before in the study of active-contour image segmentation. This newly proposed model, which we call the advection-enhanced GVF (AeGVF), is mainly inspired by the functional derivative of an adaptive total variation (TV) regularizer [12], from which we can find that the advection term indeed plays an important role in keeping the edges of the desired object. By incorporating the advection effect into the usual GVF model in solving the external force field, not surprisingly, the resulting AeGVF model is able to preserve the object boundaries efficiently. We will present several numerical examples on synthetic and real images to illustrate the high performance of the newly proposed model. Numerical experiments show that the AeGVF can effectively deal with the issues of weak edges, narrow and deep concavities, and missing edges.

The remainder of this paper is organized as follows. In Section 2, the traditional snake with GVF model and some improvements are briefly described. In Section 3, the novel AeGVF model is proposed and discussed in detail. In Section 4, some numerical experiments are performed that demonstrate the better performance of our approach compared with some existing flow models. Finally, a brief summary and conclusions are presented in Section 5.

2 Background

2.1 Traditional snakes

We consider a snake (active contour) that is a 2-D parametric curve $z(s)=(x(s),y(s))$, $s \in [0,1]$. The active contour moves through the spatial domain Ω of an image $I(x,y)$ to minimize the following energy functional over a suitable function space [14]:

$$\mathcal{E}_{snake}(z) = \int_0^1 \frac{1}{2} \Big(\alpha |z_s(s)|^2 + \beta |z_{ss}(s)|^2 \Big) + E_{ext}(z(s)) ds, \tag{2.1}$$

where the subscript s means that we differentiate a function with respect to the variable s; α and β are constant parameters controlling the snake's tension and rigidity, respectively; E_{ext} is a given external function related to the image data so that its values at the features of interest, such as the image boundaries, are as small as possible. In principle, snakes possess the continuity and smoothness characterized by the first two terms during deformation, and move toward the desired object features by the external function E_{ext}.

In order to obtain a snake $z(s)$ which at least locally minimizes the energy functional $\mathcal{E}_{snake}$, a common strategy is to use the calculus of variations and then to solve the associated Euler-Lagrange equation with appropriate boundary conditions:

$$\alpha z_{ss}(s) - \beta z_{ssss}(s) - \nabla_z E_{ext}(z(s)) = 0, \quad 0 < s < 1, \tag{2.2}$$

where ∇_z denotes the gradient operator with respect to the variable $z\!=\!(x,y)$. In practice, we will treat the curve z as a function not only in the original parameterization variable s but also in the time variable t. We then solve an evolution equation,

$$\frac{\partial z}{\partial t}(s,t)-\alpha z_{ss}(s,t)+\beta z_{ssss}(s,t)=-\nabla_z E_{ext}(z(s,t)), \quad (s,t)\in(0,1)\times(0,T], \qquad (2.3)$$

to reach a steady-state solution for a sufficiently large time $T\!>\!0$, where we have to impose the initial contour $z(s,0)\!=\!z_0(s)$ for $s\!\in\![0,1]$ with appropriate boundary conditions. When $\partial z/\partial t\!=\!0$, we obtain a steady-state solution of (2.3) which is also a solution of (2.2). Meanwhile, the evolution of the solution shows how the snake moves to the one that has the desired image attributes. In fact, (2.2) can also be viewed as a force balance equation expressed in the form

$$F_{int}(z(s))+F_{ext}(z(s))=0, \qquad (2.4)$$

where $F_{int}(z(s))\!=\!\alpha z_{ss}(s)-\beta z_{ssss}(s)$ and $F_{ext}(z(s))\!=\!-\nabla_z E_{ext}(z(s))$. Generally speaking, the internal force F_{int} suppresses the stretching and bending of the snake contour, while the external force F_{ext} attracts it to the desired image features such as the edges [32].

The external function E_{ext} is a key ingredient for the performance of image segmentation. It is usually defined as a local feature of the image, and for different images or purposes, it can be designed in different ways. For example, if an image $I(x,y)$ is a line drawing (black on white), an appropriate external function [8,14] can be chosen as one of

$$E_{ext}(x,y)=I(x,y) \quad \text{in } \Omega, \qquad (2.5)$$
$$E_{ext}(x,y)=G_\sigma*I(x,y) \quad \text{in } \Omega, \qquad (2.6)$$

where G_σ is the two-dimensional Gaussian kernel with standard deviation σ, and the symbol $*$ denotes the usual convolution. Under these circumstances, the snake will be attracted to contours with small intensities. If an image is a gray-level one and someone wants to seek step edges, two popular external functions [14] are given by

$$E_{ext}(x,y)=-|\nabla I(x,y)|^2 \quad \text{in } \Omega, \qquad (2.7)$$
$$E_{ext}(x,y)=-|\nabla(G_\sigma*I(x,y))|^2 \quad \text{in } \Omega. \qquad (2.8)$$

In these two situations, the snake will be attracted to contours with large image gradients in magnitude. Some other appropriate external functions can also be found in, e.g., [8]. We remark that the Gaussian kernel convolution can filter noises and also has a little ability to extend the capture range which is a region where the external forces are strong enough to drive contour evolution. However, increasing the value of σ to get a wide capture range is infeasible because it will weaken the external forces and blur the desired features such that the boundary location becomes more inaccurate.

From another viewpoint, a traditional external force written as the negative gradient of a potential function, namely, $F_{ext}\!=\!-\nabla_z E_{ext}$, usually has some drawbacks [32] such as (i) limited capture range: the traditional external forces only exist in small neighborhood near the image features of interest and as a consequence, the location of the initial

contour $z_0(s)$ must be put as close as to the target object. Otherwise, the snake can not move anywhere; (ii) poor convergence for concavities: a force pointing to the bottom of a concavity can pull the snake to enter the concavity, but the traditional external forces near the center of the entrance do not generally have such good property due to the limited capture range. These limitations need to be carefully considered and addressed.

2.2   The gradient vector flow model

In order to overcome the shortcomings arising in the traditional external forces, Xu and Prince [32] have proposed a novel approach to construct efficient and reliable external forces as follows. Let E_{ext} be an external function and define the so-called edge map by

$$f(x,y):=-E_{ext}(x,y) \quad \text{in } \Omega, \tag{2.9}$$

whose value is larger near the desired features. Then they considered the following minimization problem: *Find $V(x,y)=(u(x,y),v(x,y))^\top$ in a suitable function space that minimizes the energy functional*

$$\mathcal{E}(u,v) = \iint_\Omega \mu|\nabla V|^2 + |\nabla f|^2|V-\nabla f|^2 dxdy, \tag{2.10}$$

where $\mu>0$ is a regularization parameter and

$$|\nabla V| = |(\nabla u, \nabla v)^\top| = \sqrt{u_x^2 + u_y^2 + v_x^2 + v_y^2}.$$

Solution V of the minimization problem is called the gradient vector flow (GVF) field, which will be applied to replace the term $-\nabla_z E_{ext}$ in (2.2) or (2.3) to obtain a snake. A snake using the GVF as the external force will be called a GVF snake.

   In practice, one can consider the Euler-Lagrange equation associated with the minimization of (2.10) and then the GVF is obtained by solving the following initial-boundary value problem for a sufficiently large time $T>0$ to reach a steady-state solution:

$$\begin{cases} \dfrac{\partial V}{\partial t} = \mu\nabla^2 V - |\nabla f|^2(V-\nabla f) & \text{in } \Omega\times(0,T], \\ V(z,0) = \nabla f & \text{in } \Omega, \\ \nabla V\cdot n = 0 & \text{on } \partial\Omega\times(0,T], \end{cases} \tag{2.11}$$

where ∇^2 is the Laplacian operator, $\nabla V\cdot n := (\nabla u\cdot n, \nabla v\cdot n)^\top$, and n denotes the unit outer normal vector to $\partial\Omega$. From (2.10) and (2.11), we can observe that the GVF is almost equal to the traditional external force $F_{ext}=\nabla f=-\nabla E_{ext}$ when $|\nabla f|$ is sufficiently large (namely, near the edges) and the diffusion term will spread the forces to the regions far from the edges. Also, we may expect that the GVF has a wider capture range and can enter concave regions.

2.3 Some improved gradient vector flow models

As we have mentioned above, the GVF snake model may improve the ability of the traditional snake model, but it still has some limitations that should be further addressed. For example, if an image has one of the following characteristics, then the GVF snake generally exhibits a poor performance [23,29,30,33]:

(i) a weak edge and a strong edge are very close;
(ii) the image has a narrow and deep concavity;
(iii) a strong edge is near a missing edge;
(iv) the image is corrupted by noise.

To deal with these issues, there are many improved GVF fields developed and investigated in the past two decades. In the following, we briefly review some improved GVF fields that will be also carried out in our numerical experiments in Section 4 below.

2.3.1 The generalized gradient vector flow model

In [33], Xu and Prince have further developed a new external force, called the generalized gradient vector flow (GGVF) field. This GGVF is obtained by solving the following evolution problem at a sufficiently large time $T > 0$:

$$\begin{cases} \dfrac{\partial V}{\partial t} = g(|\nabla f|)\nabla^2 V - h(|\nabla f|)(V - \nabla f) & \text{in } \Omega \times (0,T], \\ V(z,0) = \nabla f & \text{in } \Omega, \\ \nabla V \cdot n = 0 & \text{on } \partial\Omega \times (0,T], \end{cases} \tag{2.12}$$

where g and h are spatially varying weighting functions depending on the gradient of the edge map f. Obviously, when $g(|\nabla f|) := \mu$ and $h(|\nabla f|) := |\nabla f|^2$, this model reduces to the original GVF model. In general, we hope that the effect of the diffusion term only exists at locations far from the edges to prevent the edges from being polluted too much and that V is equal to ∇f as much as possible when it is near the edges. To this goal, Xu and Prince [33] suggested the appropriate weighting functions as

$$g(|\nabla f|) = \exp\{-|\nabla f|/k\}, \tag{2.13}$$
$$h(|\nabla f|) = 1 - g(|\nabla f|), \tag{2.14}$$

where $k > 0$ is a constant parameter. However, both the GVF and GGVF fields still have difficulty in forcing a snake into a long and thin concavity because they mainly rely on the Laplacian operator which is an isotropic smoothing operator to produce excessive smoothness near edges.

2.3.2   The normal gradient vector flow model

Note that the Laplacian diffusion can be locally decomposed [2] as

$$\nabla^2 V = V_{TT} + V_{NN},\tag{2.15}$$

where $V_{TT} = (u_{TT}, v_{TT})^\top$ and $V_{NN} = (u_{NN}, v_{NN})^\top$ denote the second derivatives of V in the tangential direction T and normal direction N, respectively. For example,

$$u_{TT} = \frac{1}{|\nabla u|^2} \left(u_x^2 u_{yy} + u_y^2 u_{xx} - 2u_x u_y u_{xy} \right), \quad u_{NN} = \frac{1}{|\nabla u|^2} \left(u_x^2 u_{xx} + u_y^2 u_{yy} + 2u_x u_y u_{xy} \right).$$

From the viewpoint of image interpolations [7], one can observe that V_{NN} has great interpolating effects when compared with $\nabla^2 V$ and V_{TT}. Based on this observation, Ning *et al.* [23] developed a new external force, called the normal gradient vector flow (NGVF) field, which is obtained by solving the following initial-boundary value problem at a sufficiently large time $T > 0$:

$$\begin{cases} \dfrac{\partial V}{\partial t} = \mu V_{NN} - |\nabla f|^2 (V - \nabla f) & \text{in } \Omega \times (0,T], \\[2mm] V(z,0) = \nabla f & \text{in } \Omega, \\[2mm] \nabla V \cdot n = 0 & \text{on } \partial\Omega \times (0,T]. \end{cases}\tag{2.16}$$

Unlike the GGVF, the NGVF only employs the normal component V_{NN} of the $\nabla^2 V$ to generate the force fields. Thus, it is an anisotropic method and, in general, it is stable for larger time steps or larger diffusivity [23]. The anisotropic property makes the attractive forces on the bottom of the concavities have more chances to be spread out to the entrance such that the snake can be pulled into the concave regions. However, in practice, the improvement for LTIs is still limited, so the NGVF still encounters convergence issues for LTIs. In addition, the NGVF also has trouble in preventing weak-edge leakage [29].

2.3.3   The normally biased gradient vector flow model

On the other hand, focusing on the weak-edge problems, Wang *et al.* [29] proposed a new external force, called the normally biased gradient vector flow (NBGVF) field. The NBGVF is obtained by solving the following initial-boundary value problem at a sufficiently large time $T > 0$:

$$\begin{cases} \dfrac{\partial V}{\partial t} = \mu \left(V_{TT} + g(|\nabla f|) V_{NN} \right) - |\nabla f|^2 (V - \nabla f) & \text{in } \Omega \times (0,T], \\[2mm] V(z,0) = \nabla f & \text{in } \Omega, \\[2mm] \nabla V \cdot n = 0 & \text{on } \partial\Omega \times (0,T], \end{cases}\tag{2.17}$$

where $g(|\nabla f|) := \exp\{-(|\nabla f|/k)^2\}$ with the constant parameter $k > 0$. When $|\nabla f|$ is large (i.e., near the edges), this model mainly adopts the diffusion in the tangential direction

which is beneficial for preserving weak edges. When $|\nabla f|$ is getting small, the value of the bias is getting large so that this model almost uses the Laplacian diffusion to carry forces in homogeneous regions. This approach can preserve weak edges efficiently but do not have the ability for moving into LTIs [29]. Although one may design a weighted function g whose values are bigger than 1 in homogeneous regions such that the effect of V_{NN} becomes larger, it may cause the numerical instability so that the time step should be taken carefully in the numerical discretization scheme.

2.3.4 An adaptive diffusion flow model

To have both good properties of weak edge preserving and concavity convergence, Wu *et al.* [30] proposed an adaptive diffusion flow (ADF), which is obtained by solving the following initial-boundary value problem at a sufficiently large time $T > 0$:

$$
\begin{cases}
\dfrac{\partial V}{\partial t} = g(|\nabla f|)\left\{\gamma\dfrac{\Delta_\infty V}{|\nabla V|^2} + (1-\gamma)\nabla\cdot\left(\dfrac{\Phi'(|\nabla V|)}{|G_\sigma * \nabla V|}G_\sigma * \nabla V\right)\right\} \\
\qquad\qquad -h(|\nabla f|)\big(V-\nabla f\big) & \text{in } \Omega\times(0,T], \\
V(z,0)=\nabla f & \text{in } \Omega, \\
\nabla V\cdot n = 0 & \text{on } \partial\Omega\times(0,T],
\end{cases}
\tag{2.18}
$$

where g and h are the same as those of the GGVF, Δ_∞ denotes infinity Laplacian (cf. [1]),

$$
\Phi(|\nabla V|) = \frac{1}{p(|\nabla f|)}\left(\sqrt{1+|G_\sigma * \nabla V|^2}\right)^{p(|\nabla f|)}, \tag{2.19}
$$

$$
p(|\nabla f|) = 1 + \frac{1}{1+|\nabla(G_\sigma * f)|}, \tag{2.20}
$$

$$
\gamma = \begin{cases} \left(1 - f^2/(5k^2)\right)^2 & \text{if } (f^2/5)\le\ell^2, \\ 0 & \text{otherwise,} \end{cases} \tag{2.21}
$$

$$
\ell = 1.4826\big(E(|\nabla f - E(|\nabla f|)|)\big), \tag{2.22}
$$

with $k > 0$ a constant parameter and $E(\cdot)$ the mean value. One can find that when $\gamma = 0$, the smoothness is determined by the harmonic hypersurface functional, which has the ability to protect weak edges. On the other hand, when $\gamma = 1$, the smoothness is dominated by the infinite Laplacian term, which has the ability to converge into narrow and deep concavities [30]. With this adaptive diffusion strategy, the ADF is expected to possess the advantages of both models of the NGVF and NBGVF fields.

3 The advection-enhanced gradient vector flow model

From the discussions in the previous section, it can be concluded that the weighting functions g and h, together with the diffusion in normal direction V_{NN}, is able to improve the

convergence performance for an image having LTIs, while the diffusion in tangential direction V_{TT} can preserve weak edges. Indeed, many modified GVF models in the literature are further developed based on the direct combination of them. In this section, we are going to introduce a novel GVF model built on a new concept for coping with weak and missing edges. In principle, the ingredients of our new GVF model should include the normal diffusion V_{NN}, the weighting functions g and h, and further we desire a new alternative to the tangential diffusion V_{TT} for better preserving the vector fields of external force along image edges.

3.1 Advection: an alternative to tangential diffusion

In image denoising, it is well known that the total variation (TV) regularization model [24], whose regularizer is chosen as the L^1 norm of the image gradient, can suppress the variation caused by noise without penalizing the edge gradient too much. Using the bounded variation space, the TV model successfully preserves image features such as edges or other sharp structures. To develop an alternative to V_{TT}, we begin with, borrowing the idea from image denoising, the following adaptive TV regularizer [12]:

$$\iint_\Omega \alpha |\nabla w| \, dx dy, \tag{3.1}$$

where $\alpha := \alpha(x,y)$ is used as a spatial-varying controller to improve the performance of the original TV regularizer. Here w is a scalar function in a suitable function space of interest defined on $\overline{\Omega}$, which can be viewed as either component of the vector field $V = (u,v)^\top$ discussed in Section 2; that is, one can take w as u or take w as v.

By a direct computation, the associated functional derivative of (3.1) is given by

$$
\begin{aligned}
&-\nabla \cdot \left(\frac{\alpha}{|\nabla w|} \nabla w \right) \\
={}&-\frac{\alpha}{|\nabla w|} \nabla^2 w - \nabla \left(\frac{\alpha}{|\nabla w|} \right) \cdot \nabla w \\
={}&-\frac{\alpha}{|\nabla w|} \nabla^2 w - \left(\frac{\nabla \alpha}{|\nabla w|} - \frac{\alpha}{|\nabla w|^3} \left(w_x w_{xx} + w_y w_{yx}, \ w_x w_{xy} + w_y w_{yy} \right)^\top \right) \cdot \nabla w \\
={}&-\frac{\alpha}{|\nabla w|} \nabla^2 w - \frac{1}{|\nabla w|} \nabla \alpha \cdot \nabla w + \frac{\alpha}{|\nabla w|^3} \left(w_x^2 w_{xx} + 2 w_x w_y w_{yx} + w_y^2 w_{yy} \right) \\
={}&-\frac{\alpha}{|\nabla w|} \nabla^2 w - \frac{1}{|\nabla w|} \nabla \alpha \cdot \nabla w + \frac{\alpha}{|\nabla w|} w_{NN} \\
={}&-\frac{1}{|\nabla w|} \left(\alpha w_{TT} + \nabla \alpha \cdot \nabla w \right),
\end{aligned}
\tag{3.2}
$$

which reveals that both the terms αw_{TT} and $\nabla \alpha \cdot \nabla w$ are closely related to the ability for preserving image edges. In particular, if we take $\alpha \equiv 1$, then $\alpha w_{TT} = w_{TT}$ and $\nabla \alpha \cdot \nabla w$ vanishes. In this case, the regularizer in (3.1) reduces to the original TV and the associated

operator obtained from the functional derivative is simply the tangential diffusion operator w_{TT}. This operator w_{TT} in general has a global support and it will form a complete diffusion operator when combining with the normal diffusion operator w_{NN}, which probably leads to the failure for the snake entering LTIs. Intuitively, if we can choose some appropriate controller α such that these two terms αw_{TT} and $\nabla \alpha \cdot \nabla w$ in (3.2) have only local support near edges then we can construct a new operator which preserves edges while leaves other regions unaffected. To this end, an appropriate choice is to take $\alpha = f$, where f is the edge map defined in (2.9), and the two resulting terms operate locally as desired. Since the value of f is large at edges and small otherwise, the diffusion term $f w_{TT}$ works almost on edges, that means, $f w_{TT}$ only preserves vector fields on edges. However, what we want to preserve are the vector fields beside edges rather than the vector fields on edges, since they may extrude the snake. To accurately stop the snake at edge locations, the force fields on edges should be removed. As a consequence, the term αw_{TT} in (3.2) with $\alpha = f$ has no substantial effect and can be discarded, and the advection term $\nabla \alpha \cdot \nabla w$ becomes $\nabla f \cdot \nabla w$.

To summarize, minimizing the proposed regularizer in (3.1) is expected to effectively preserve edges. Taking α to be the edge map f, the resulting advection term $\nabla f \cdot \nabla w$ can be regarded as a better edge-preserving term than the tangential diffusion term w_{TT}, since it can not only increase the strength of attractive forces to prevent the data fidelity from being destroyed but also leave the homogeneous regions unaffected. From this perspective, we now therefore propose a heuristic model, called the advection-enhanced GVF (AeGVF), by solving the following evolution problem at a sufficiently large time $t = T$ to approach a steady-state solution:

$$\begin{cases} \dfrac{\partial V}{\partial t} = g(|\nabla f|) V_{NN} - \left(\mu_c \chi_{\Omega \setminus E_c} + \mu_p \chi_{E_p} \right) \nabla f \cdot \nabla V \\ \qquad\quad - h(|\nabla f|) \left(V - \nabla f \right) & \text{in } \Omega \times (0,T], \\ V(z,0) = \nabla f & \text{in } \Omega, \\ \nabla V \cdot n = 0 & \text{on } \partial \Omega \times (0,T], \end{cases} \quad (3.3)$$

where $g(|\nabla f|) := \exp\{-(|\nabla f|/k)^2\}$, $h(|\nabla f|) := 1 - g(|\nabla f|)$, μ_c and μ_p are two nonnegative parameters, $\nabla f \cdot \nabla V := (\nabla f \cdot \nabla u, \nabla f \cdot \nabla v)^\top$, the characteristic function χ_A of a given set A is defined by

$$\chi_A(z) = \begin{cases} 1 & \text{if } z \in A, \\ 0 & \text{if } z \notin A, \end{cases} \quad (3.4)$$

E_c and E_p denote the sets of corner points and endpoints of boundary curves in the image, respectively. More precisely, E_c collects points on boundary curves where gradient is not well defined, while E_p collects points on the two ends of boundary curves which are not closed. In the next subsection, we will give details of the AeGVF model and further discuss the effect of the advection term $\nabla f \cdot \nabla V$.

3.2 Details about the AeGVF model

To complete the AeGVF model (3.3), we begin with the following simple setting:

$$\mu_c > 0, \quad \mu_p = 0, \quad E_c = \varnothing. \tag{3.5}$$

In other words, we add the advection term $\nabla f \cdot \nabla V$ in the whole domain and then use the parameter μ_c to control its strength. Numerical results of a simple test are shown in Fig. 1 for three different values of μ_c. From the numerical results, one can find that, as μ_c increases, the missing edges are getting recovered, especially the right missing edge which is located very close to a strong edge. Although the recovery can be made by increasing the strength of advection, there are two undesirable results that should be noticed. First, as μ_c is getting larger, the force fields near the corner points may gradually be destroyed. Second, for each missing edge, the attractive forces from the two endpoints may sometimes be cancelled out at the midpoint, making the snake be pulled out to the strong edges along a horizontal line, see Fig. 1(e).

To deal with the first undesirable result, we use the Harris corner detector with the MATLAB built-in command corner(f,'SensitivityFactor',0.2) to find corner points, where f is the given edge map, and then remove the advection effect at these corner points. The collection of these corner points is denoted by the set E_c. In fact, the gradient of a two-variable function is determined uniquely provided that the function is differentiable. So, theoretically, ∇f is not well defined at corner points of an image. For this reason, we should exclude the advection effect at image corners.

To deal with the second undesirable result, we use the MATLAB built-in command bwmorph(skellmage, 'Endpoints') to find endpoints of edges and then the advection terms are randomly imposed on half pixels in the 3×3-pixel neighborhood of each endpoint to disturb the symmetry of the attractive forces. The randomness used here is a discrete uniform distribution which is especially good for dealing with symmetric structures near the endpoints. The collection of those random pixels to impose advection terms is denoted by the set E_p.

With these two auxiliary sets E_c and E_p, the improvement can be clearly observed in Example 4.1 below, where the missing edges are recovered completely.

3.3 Numerical implementation of snake models

In this subsection, let us describe some preliminaries needed in the numerical implementations of various snake models.

- We use the explicit forward Euler difference scheme in time and the centered difference scheme in space, so the models with the weighting function g must satisfy the stability condition [33],

$$\Delta t \le \frac{\Delta x \Delta y}{4 g_{max}}, \tag{3.6}$$
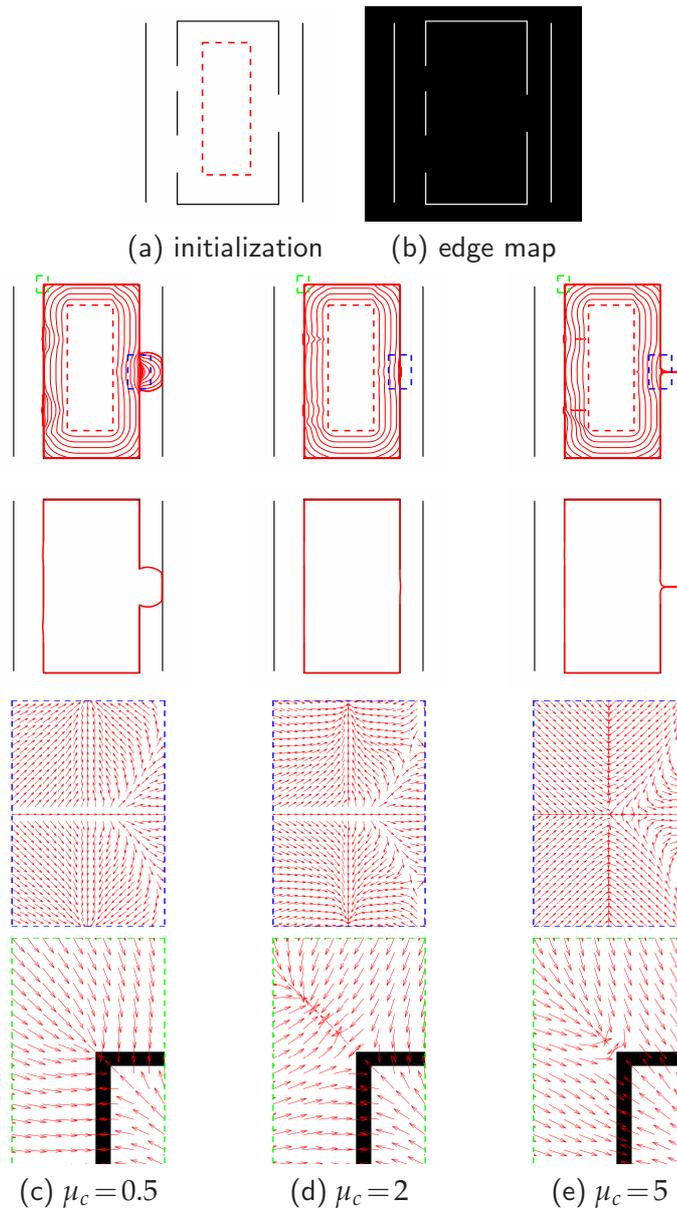
Figure 1: Numerical results of the AeGVF model for three different μ_c with $\mu_p=0$, $E_c=\varnothing$ in all cases, where the top row illustrates the initialization and edge map; the second row to the last row show the evolution processes, the segmentation results, the external force fields of blue and green zoomed-in regions, respectively.

where Δx and Δy are spatial mesh sizes in x- and y-direction, respectively, and g_{max} is the maximum value of g. For all examples in Section 4 below, we always take $\Delta x = \Delta y = 1$ and set $\Delta t = 0.25$ for all improved GVF models.

- Following the idea that the filter G_σ can helpfully suppress the influence of noise [30], we therefore use $G_\sigma * V$ instead of V in each time step for all improved GVF models, where σ depends on the noise level of an image. For a noise-free image, σ can be taken very small. When the noise level is getting higher, σ should be taken larger. Taking the AeGVF as an example, in each time step we will approximate $V^{(m)}$ by using the centered difference scheme,

$$
\begin{cases}
V^{(m)} = G_\sigma * V^{(m-1)} + \Delta t g(|\nabla f|)(G_\sigma * V^{(m-1)})_{NN} \\
\qquad - \Delta t \left(\mu_c \mathcal{X}_{\Omega \setminus E_c} + \mu_p \mathcal{X}_{E_p} \right) \nabla f \cdot \nabla (G_\sigma * V^{(m-1)}) \\
\qquad - \Delta t h(|\nabla f|)\left(G_\sigma * V^{(m-1)} - \nabla f \right) & \text{in } \Omega, \quad \text{for } m \geq 1, \qquad (3.7) \\
V^{(0)} = \nabla f & \text{in } \Omega, \\
\nabla V^{(m)} \cdot n = 0 & \text{on } \partial\Omega, \quad \text{for } m \geq 1.
\end{cases}
$$

- Finally, we remark that to accurately stop the snake at the edge locations, intuitively, there should be no force fields on edges. So we remove all the force fields on edges for synthetic images in our AeGVF snake, that is, we substitute $V\mathcal{X}_{\{f=0\}}$ for V and then solve the following modified snake model for synthetic images:

$$
\frac{\partial z}{\partial t}(s,t) - \alpha z_{ss}(s,t) + \beta z_{ssss}(s,t) - V\mathcal{X}_{\{f=0\}} = 0, \quad (s,t) \in (0,1) \times (0,T], \qquad (3.8)
$$

with suitable initial contour and boundary conditions for $0 \leq t \leq T$.

4 Numerical experiments

In this section, we are going to compare the performance of various improved GVF models, including the GGVF, NGVF, NBGVF, ADF and the newly proposed AeGVF, on several desirable abilities such as missing edge recovery, convergence for LTIs, prevention of weak-edge leakage and noise robustness for both synthetic and real images. Numerical results will demonstrate the high performance of the newly proposed AeGVF model. We remark that all the edge maps f are normalized to $[0,1]$, and the parameters α, β and σ will be described clearly in each example.

Example 4.1 (Missing edge recovery). In this example, we consider two images respectively with U-shaped and rectangle objects to study the efficiency of the proposed AeGVF snake for recovering missing edges. We take the edge map $f(x,y) = -I(x,y)$, where $I(x,y)$ is the given image.

(i) For the U-shaped image, we set $\alpha = \beta = 0.5$, and $\sigma = 0.15$. Fig. 2 shows that all the test models are able to recover the missing edge very well, and the AeGVF snake
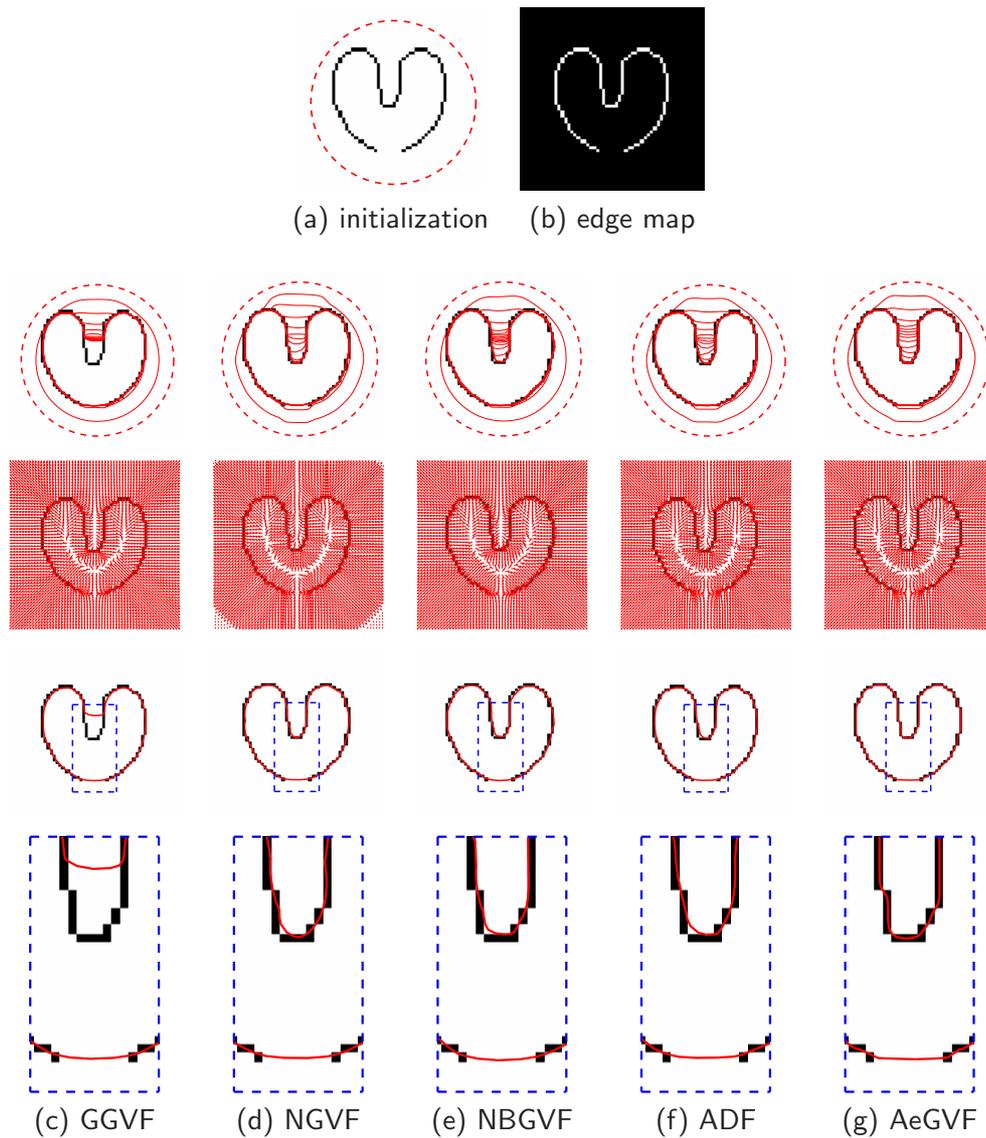
Figure 2: Numerical results of some improved GVF models for Example 4.1, where the top row illustrates the initialization and edge map; the second row to the last row show the evolution processes, the external force fields, the segmentation results and the zoomed-in regions, respectively. In all improved models, we set $\Delta t = 0.25$, $\sigma = 0.15$, and $k = 0.01$ in the weighting function g. Column (c) shows the results of the GGVF snake; (d) is for the NGVF snake with $\mu = 0.2$; (e) is for the NBGVF snake with $\mu = 0.5$; (f) is for the ADF snake; (g) is for the AeGVF snake with $\mu_c = \mu_p = 0.01$.

locates the boundary perfectly; see the zoomed-in regions in Fig. 2. In other words, removing all the force fields on edges for the line-drawing image can make snakes stay on edges accurately.
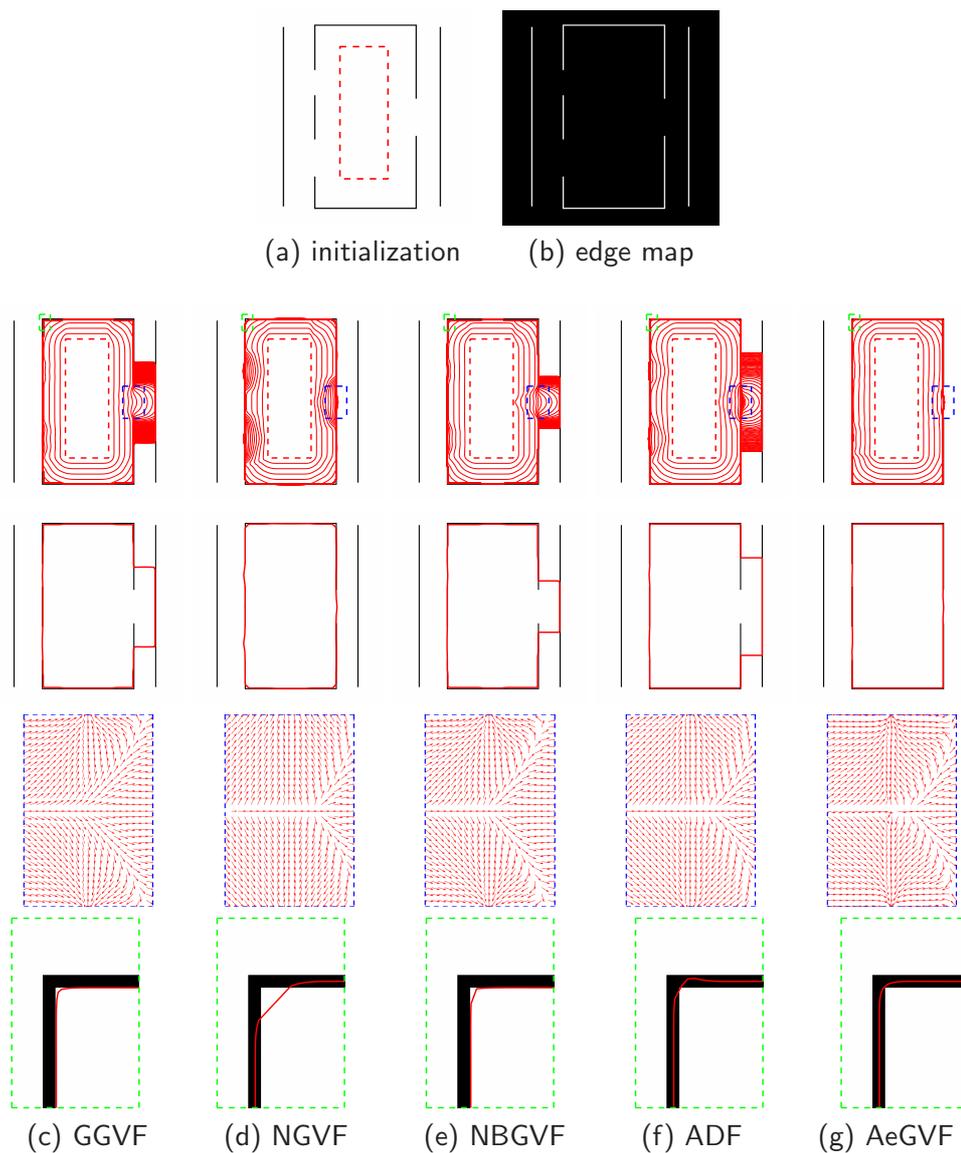
Figure 3: Numerical results of some improved GVF models for Example 4.1, where the top row illustrates the initialization and edge map; the second row to the last row show the evolution processes, the segmentation results, the external force fields in the blue zoomed-in regions and the segmentation results in the green zoomed-in regions, respectively. In all improved models, we set $\Delta t = 0.25$, $\sigma = 0.01$, and $k = 0.01$ in the weighting function g. Column (c) shows the results of the GGVF snake; (d) is for the NGVF snake with $\mu = 0.5$; (e) is for the NBGVF snake with $\mu = 0.5$; (f) is for the ADF snake; (g) is for the AeGVF snake with $\mu_c = \mu_p = 2$.

(ii) For the image of a rectangle object, we set $\alpha = \beta = 0.5$, and $\sigma = 0.15$. In this case, we put two strong edges nearby. The distance between the left strong edge and the rectangle is of 25-pixel width, and the other is of 19-pixel width. The segmentation results are shown in Fig. 3. It can be seen that the AeGVF snake attaches to the

object very well. Not only the sharp corners are built correctly, but also the missing parts are reconstructed robustly even though the strong edges are very close to the missing ones. With a closer observation, one can find that the evolution process of the AeGVF is smooth near the missing edges without struggling too much. In addition, the force fields (see the blue zoomed-in regions in Fig. 3) clearly show that the attractive forces from the up and down endpoints are more concentrated than those of the other snakes such that the gradient vectors along the horizontal line can be cut off. This phenomenon demonstrates how the advection effects can indeed improve the ability of recovering missing edges.

In summary, these numerical results show that the AeGVF snake has much better performance than the other snakes in terms of recovering the missing parts.

Example 4.2 (Convergence to LTIs). In this example, we consider a narrow and deep concavity which is 3 pixels in width and 150 pixels in depth and it is often used to test the convergence performance of an improved snake for LTIs [30]. The same concavity with inhomogeneous intensity and missing edges are also considered to conduct more delicate comparisons. Again, in this example, we take the edge map $f(x,y) = -I(x,y)$.

(i) First, we consider homogeneous edge with intensity zero, and there is no missing edge imposed. In this case, we set $\alpha = 0.1$, $\beta = 0.06$ and $\sigma = 0.15$. The experimental results are given in Fig. 4, which shows that both the ADF and AeGVF snakes successfully converge to the bottom of LTI, while the others stop at the entrance. Observing further the force fields in the zoomed-in regions, one can find that the gradient vectors of both the ADF and AeGVF point straight downward to the bottom of the concavity, while the others do not. Finally, we note that the AeGVF snake attaches to the boundary much better than the ADF snake.

(ii) Second, to further distinguish the convergence properties of the ADF and AeGVF snakes, we consider another two cases of inhomogeneous edge intensity with bottom edges partly and fully removed. In both two cases, we set $\alpha = 0.1$, $\beta = 0.2$ and $\sigma = 0.15$. The experimental results are given in Fig. 5. From Fig. 5(c) and Fig. 5(d), we can find that the AeGVF snake correctly converges to the target boundary with 450 iterations for generating force fields, but the ADF snake can not reach a similar result even with 550 iterations. In the second case, we additionally remove the bottom edge of the concavity, see Fig. 5(b), which signifies that the downward vectors can not come from the bottom edge but should come from the entrance instead. From Fig. 5(e), one can see that the ADF produces force fields pointing upward, which stop the snake moving into the concavity. However, the AeGVF preserves the symmetry of the force fields to keep the force propagation going downward, which makes the snake converge to the desired object almost perfectly. So, the proposed AeGVF snake may converge faster and more robustly than the ADF snake for entering a narrow and deep concavity.

(a) initialization   (b) edge map

(c) GGVF   (d) NGVF   (e) NBGVF   (f) ADF   (g) AeGVF
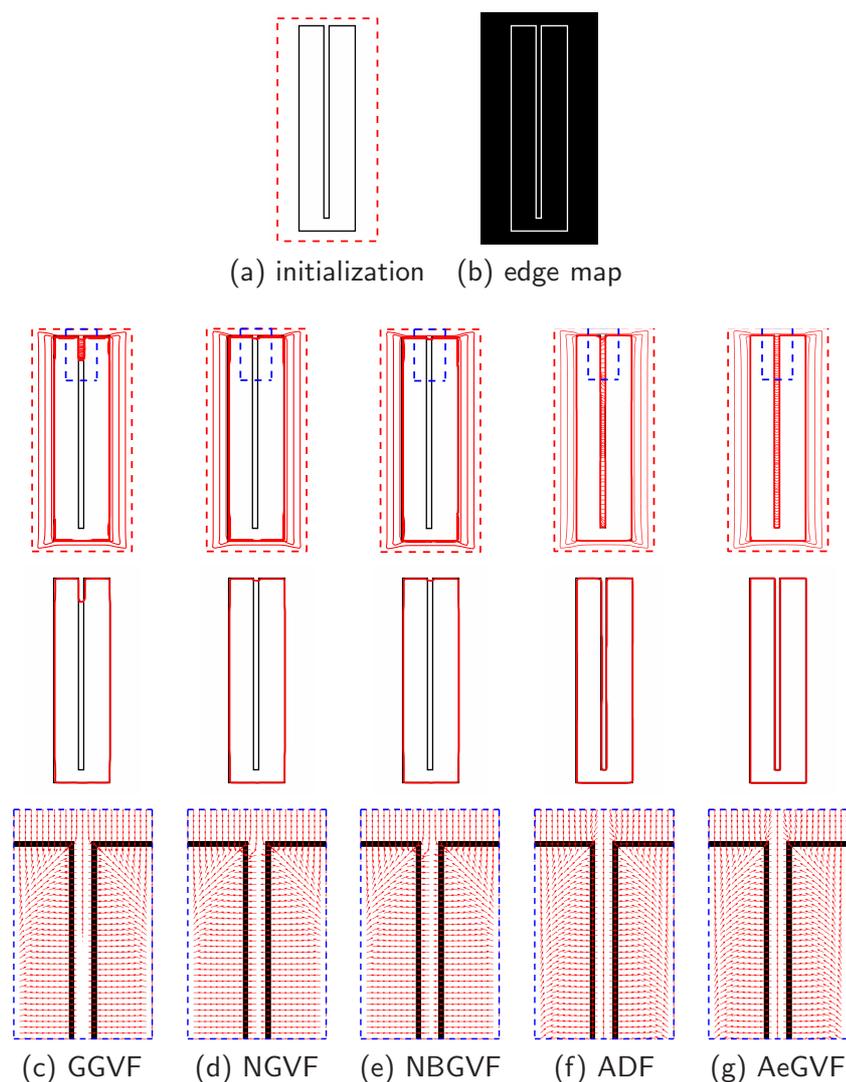
Figure 4: Numerical results of some improved GVF models for Example 4.2, where the top row illustrates the initialization and edge map; the second row to the last row show the evolution processes, the segmentation results, and the external force fields in the blue zoomed-in regions, respectively. In all improved models, we set $\Delta t = 0.25$ and $\sigma = 0.15$. Column (c) shows the results of the GGVF snake with $k = 0.005$; (d) is for the NGVF snake with $\mu = 0.2$; (e) is for the NBGVF snake with $k = 0.01$ and $\mu = 0.2$; (f) is for the ADF snake with $k = 0.005$; (g) is for the AeGVF snake with $k = 0.01$, $\mu_c = 0.01$ and $\mu_p = 0$.

Example 4.3 (Preventing weak-edge leakage). In this example, we consider four circular-shaped objects which are commonly used to test the ability of a snake for preventing weak-edge leakage. In this example, we take the edge map $f(x,y) = |\nabla I(x,y)|$.

   (i) The first two images possess small blurry regions around the upper left boundary [27, 30, 31], see Fig. 6(a)-(b). The image intensity on the blurry regions changes so
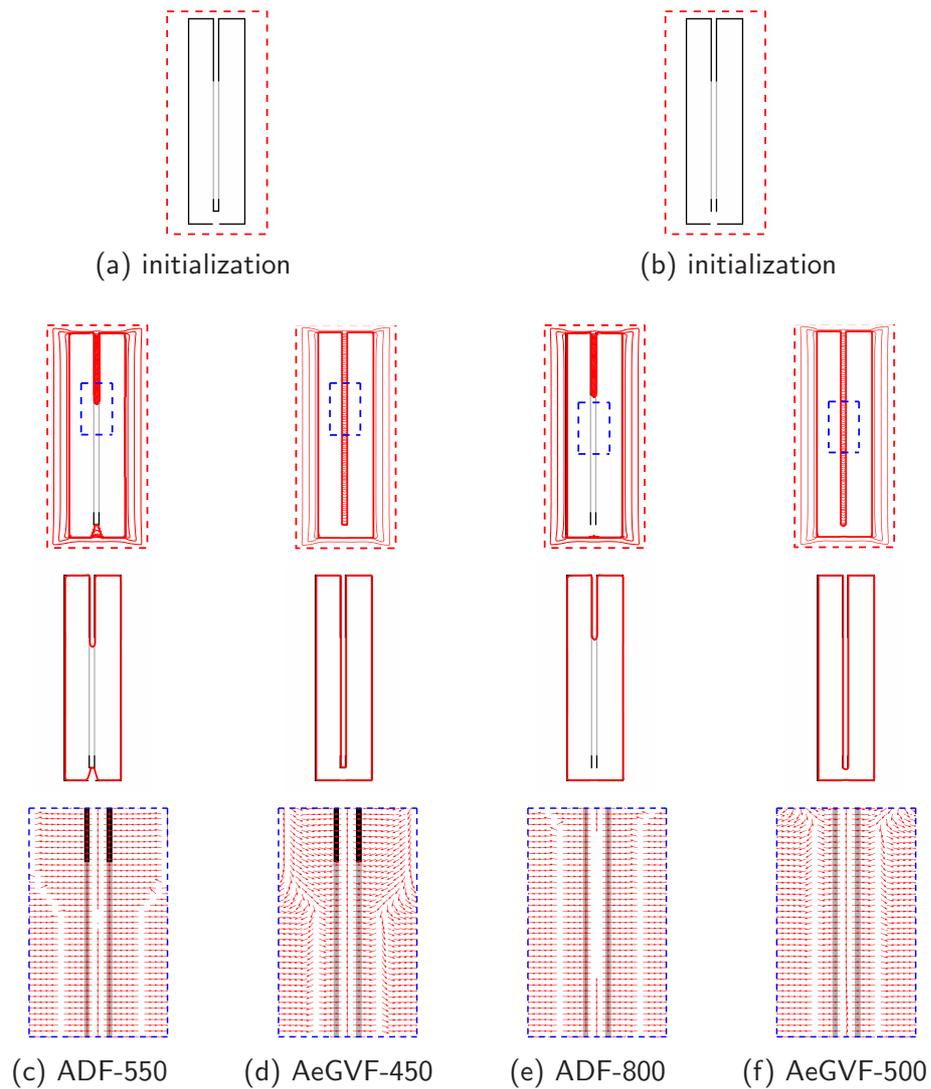
Figure 5: Numerical results of the ADF and AeGVF models for Example 4.2, where the top row illustrates the initializations; the second row to the last row show the evolution processes, the segmentation results, and the external force fields in the blue zoomed-in regions, respectively. The difference between (a) and (b) comes from the existence of the bottom edge of the concavity. Columns (c) and (d) show the results of the ADF and AeGVF snakes for the case (a), respectively. The dash sign is followed by the iteration numbers for generating force fields. Columns (e) and (f) are for the case (b). In both cases, $\Delta t = 0.25$, $\sigma = 0.15$, $k = 0.005$ for the ADF snake, and $k = 0.01$, $\mu_c = \mu_p = 2$ for the AeGVF snake.

slowly that the corresponding edge maps will not have clear boundary indication such that weak-edge leakage may occur there. In both cases, the parameters are chosen as $\alpha = 0.5$, $\beta = 0.5$, $\Delta t = 0.25$ and $\sigma = 0.01$. The experimental results show that the NGVF snake leaks from the weak edges in both cases, while the GGVF snake

(a) initialization   (b) initialization

(c) GGVF   (d) NGVF   (e) NBGVF   (f) ADF   (g) AeGVF
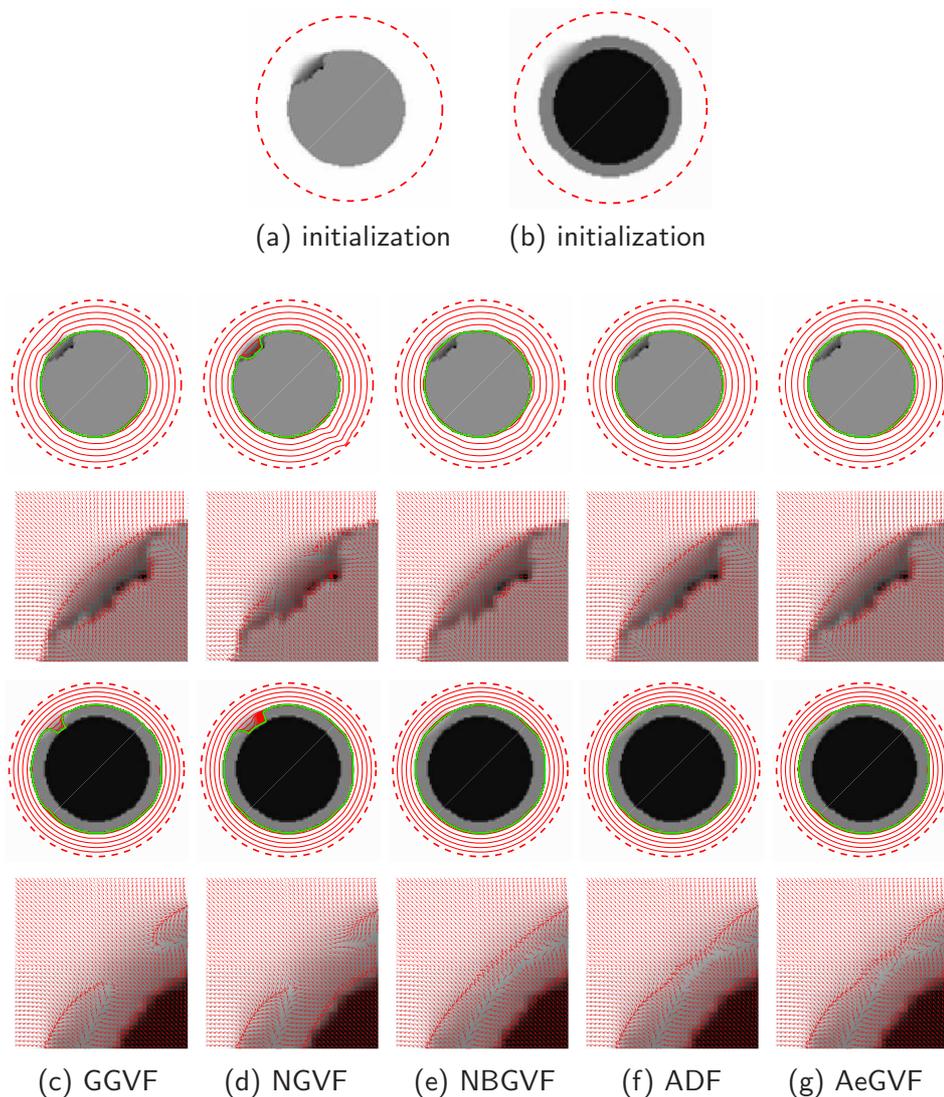
Figure 6: Numerical results of some improved GVF models for Example 4.3, where the top row illustrates the initializations for each tested image; the second row and the fourth row show the evolution processes of each improved snake for the two images, respectively; the third row and the fifth row are the external force fields near the blurry regions. The green curves represent the segmentation results. In all improved models, we set $\Delta t = 0.25$ and $\sigma = 0,01$ for (a) and (b). Column (c) shows the results of the GGVF snake with $k = 0.02$; (d) is for the NGVF snake with $\mu = 0.2$; (e) is for the NBGVF snake with $k = 0.001$ and $\mu = 0.2$; (f) is for the ADF snake with $k = 0.02$; (g) is for the AeGVF snake with $k = 0.01$, $\mu_c = 2$, $\mu_p = 0$.

preserve the weak edge in the first case. In contrast to the strenuous preservation of the GGVF snake, it can be observed from the evolution processes and the force fields that the NBGVF, ADF, and AeGVF snakes all preserve weak edges easily and have very good performances even when a strong edge is nearby in the second case.
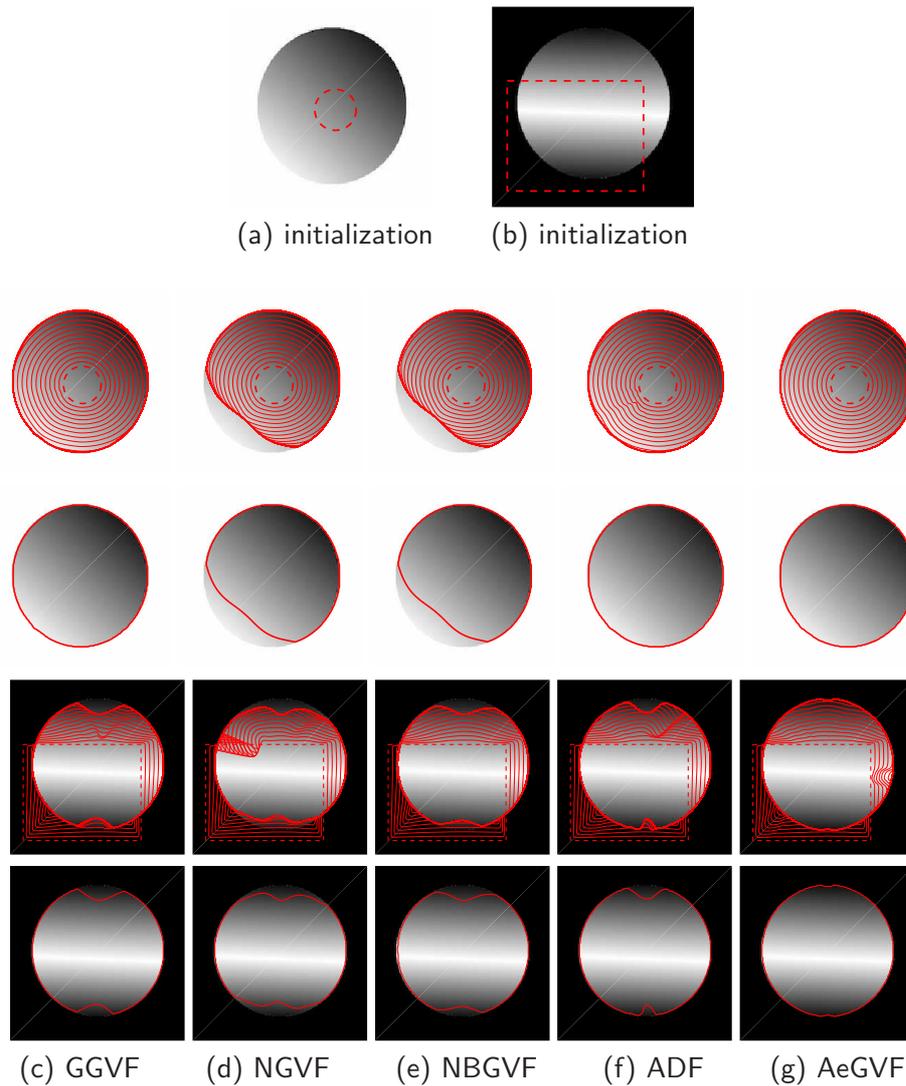
Figure 7: Numerical results of some improved GVF models for Example 4.3, where the top row illustrates the initializations for each tested image; the second row and the fourth row show the evolution processes of each improved snake for the two images; the third row and the fifth row are the segmentation results. In all improved models, we set $\Delta t = 0.25$ and $\sigma = 1$ and 0.01 for (a) and (b), respectively. Column (c) shows the results of the GGVF snake with $k = 0.05$ and 0.8 for (a) and (b), respectively; (d) is for the NGVF snake with $\mu = 0.2$ and 0.8; (e) is for the NBGVF snake with $k = 0.02$, $\mu = 0.2$ and 0.8; (f) is for the ADF snake with $k = 0.05$ and 0.8; (g) is for the AeGVF snake with $k = 0.02$, $\mu_c = 2$, $\mu_p = 0$.

(ii) We also consider another two images with gradient colors [3, 30], see Fig. 7(a)-(b). The parameters are set as $\alpha = 0.5$, $\beta = 0.5$, and $\Delta t = 0.25$ for both cases, $\sigma = 1$ for the first case (a), and $\sigma = 0.01$ for the second case (b). The lower left part of the first image is relatively bright so the boundary is unclear. The segmentation results are shown in Fig. 7. The NGVF and NBGVF snakes converge to almost the same wrong

boundary while the others converge correctly. The second image possesses shady areas on the top and bottom of the circular object. The experimental results show that the AeGVF snake converges to an acceptable boundary shape while the others converge falsely to apple-shaped boundaries. These two test images demonstrate the good performance of the AeGVF snake for protecting weak edges caused by inhomogeneous illumination.

Example 4.4 (Testing on noisy images). Active-contour snake is sometimes sensitive to noise, so we are going to study the noise robustness of the AeGVF snake in this example.

(i) First, we consider three different types of noises on the U-shaped object [30, 32], including the salt-and-pepper noise, speckle noise and Gaussian noise. The noises are created by MATLAB built-in functions imnoise(I, 'salt & pepper', 0.08), imnoise(I, 'speckle', 0.08) and imnoise(I, 'gaussian', 0.0001). The experimental comparisons are shown in Fig. 8. In all cases, we set $\alpha = 0.5$, $\beta = 0.5$, and $\Delta t = 0.25$. The edge map is taken as $f(x,y) = -I(x,y)$ with no filter here. The parameter σ for different kinds of noise and improved models are described in the caption of Fig. 8. Comparing the evolution processes, one can observe that the GGVF, NGVF and NBGVF snakes are more sensitive to the noises even though the noises are only with low percentages. In contrast, both the ADF and AeGVF snakes achieve satisfactory segmentation results, and there is a notable difference in the evolution processes between the two snakes. The AeGVF snake seems more stable in the evolution than the ADF snake for the speckle noise case, and is comparable to the ADF snake for the other cases.

(ii) To further examine the noise sensitivity of the AeGVF snake, we added much stronger impulse noises to a harmonic curve [31, 33] parametrized by

$$C(r,t) = (r\cos t, r\sin t), \tag{4.1}$$

where $r = 1 + 0.23\cos 4t$, $t \in [0, 2\pi]$. The experimental results for different noise densities are shown in Fig. 9. In all cases, we set $\alpha = 0.5$, $\beta = 0.5$, and $\Delta t = 0.25$. The edge map is $f(x,y) = |\nabla(G_{\sigma_f} * I(x,y))|$ with $\sigma_f = 2$. The various noise densities of salt-and-pepper noise are presented in the caption of Fig. 9. The results show that the performance of the AeGVF snake is quite robust. Although the segmentation result may be a little inaccurate when the noise density is bigger than 50%, it is still acceptable since the use of filter to smooth noise for the edge map may blur the boundary so that the snake deviates the correct boundary position of the object in an unavoidable way.

Example 4.5 (A strategy for choosing μ_c and μ_p). In this example, we give a strategy to choose appropriate values for the nonnegative parameters μ_c and μ_p. From the discussion in Subsection 3.1 and the numerical results reported in Examples 4.1–4.4, we know that the advection effect is mainly used to preserve image edges or sharp structures. In other words, if our purpose is to (i) protect a weak or missing edge from being passed

(a) salt-and-pepper   (b) speckle   (c) Gaussian

(d) GGVF   (e) NGVF   (f) NBGVF   (g) ADF   (h) AeGVF
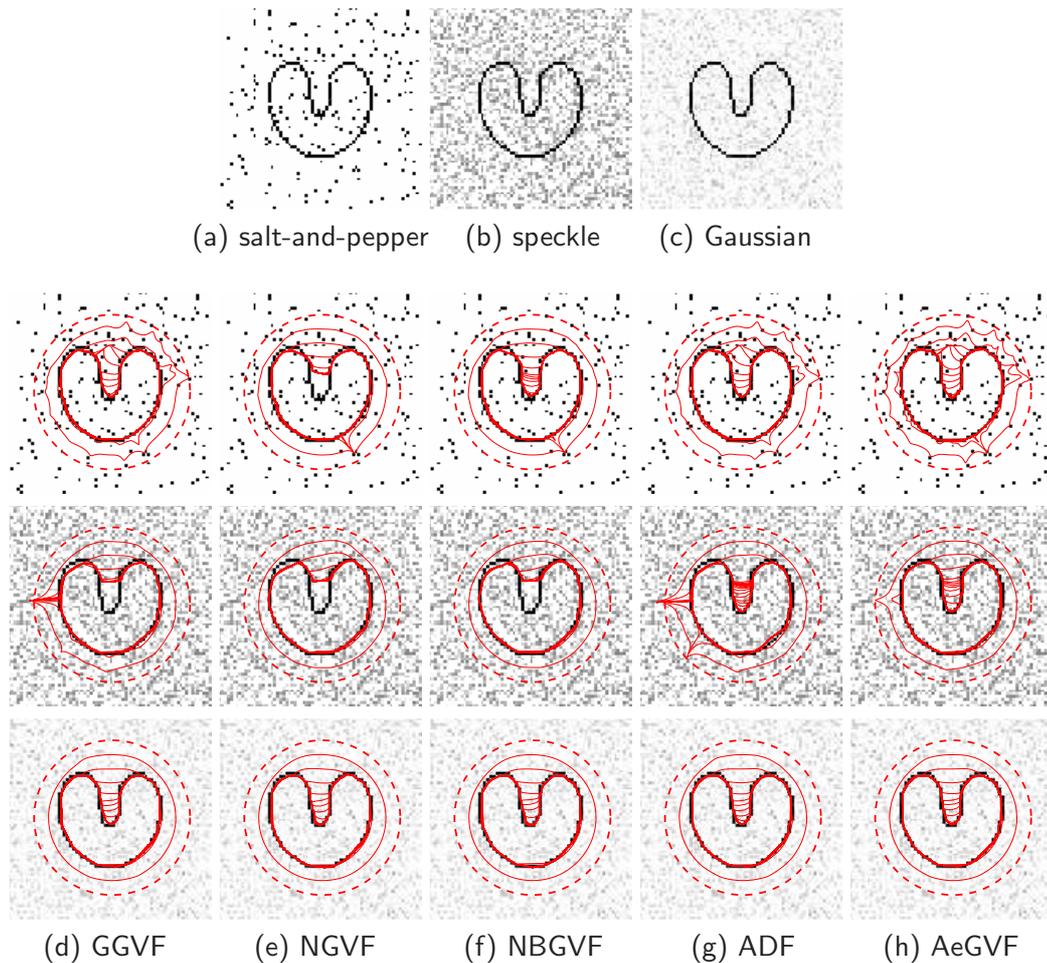
Figure 8: Numerical results of some improved GVF models for Example 4.4, where three types of noises (salt-and-pepper, speckle, Gaussian) on the U-shaped image are listed in (a)–(c). In all improved models, we set $\Delta t = 0.25$. Column (d) shows the results of the GGVF snake with $(k,\sigma) = (0.01,1.5)$, $(1,1)$, and $(0.5,1)$ for the top (salt-and-pepper), middle (speckle) and down (Gaussian) cases, respectively. Similarly, (e) is for the NGVF snake with $(\mu,\sigma) = (0.2,0.6)$, $(0.2,0.5)$, $(0.2,0.5)$; (f) is for the NBGVF snake with $(\mu,k,\sigma) = (0.2,0.01,0.6)$, $(0.2,0.5,0.5)$, $(0.2, 0.2, 0.5)$; (g) is for the ADF snake with $(k,\sigma) = (0.01,1.5)$, $(2,1)$, $(0.5,1)$; (h) is for the AeGVF snake with $\mu_p = 0$, $(\mu_c,k,\sigma) = (2,0.01,1.5)$, $(5,0.5,1)$, $(2,0.2,1)$.

through by snakes (cf. Figs. 3, 5, 6, 7); or (ii) keep the edge strength from being destroyed by a large σ (cf. Fig. 7, 8, 9), then we may choose a relatively large value for μ_c and μ_p. On the other hand, if there is no such necessity (cf. Figs. 2, 4), we may choose a smaller value for μ_c and μ_p. Indeed, the choices of μ_c and μ_p in Examples 4.1–4.4 obey such a principle and the numerical results are rather satisfactory. Moreover, we also examine the sensitivity of the AeGVF model with respect to the advection parameters for the images that meet (i) or (ii); see Fig. 10. From the numerical results depicted in Fig. 10, we can find that the image with a long and thin concavity has a smaller range of values for μ_c and
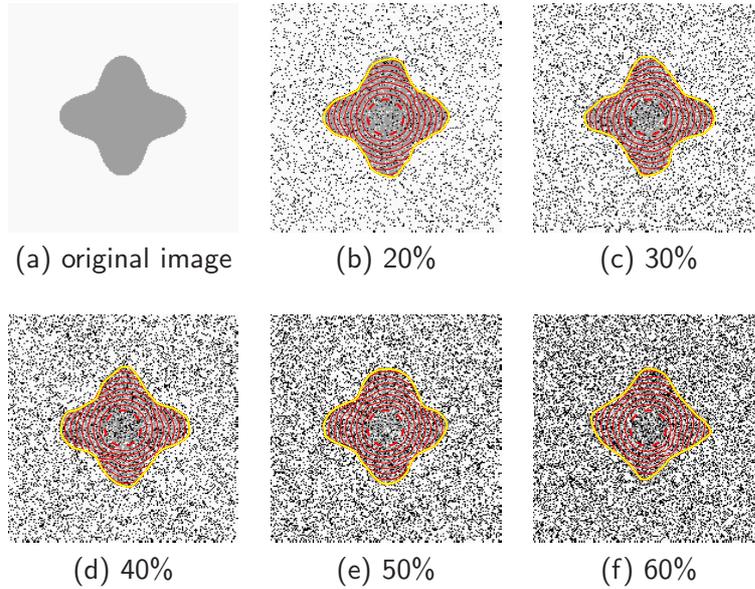
Figure 9: Numerical results of the AeGVF model for Example 4.4, where a simple harmonic curve is polluted with various levels of salt-and-pepper noises. (a) is the original image; (b)–(f) are the evolution processes of the AeGVF snake for noise densities 20%–60%. The corresponding settings for (k,σ) are taken as $(0.1,2)$, $(0.1,2)$, $(0.05,2)$, $(0.03,2)$, and $(0.02,4.5)$, respectively. In all cases, $\Delta t = 0.25$, $\mu_p = 0$ and $\mu_c = 2$. The yellow curves depict the segmentation results.

μ_p (about $0.8 \sim 2.8$) that makes the snake converge to the object successfully. However, overall, the AeGVF snake is rather robust with respect to the advection parameters μ_c and μ_p.

Finally, we close this section by applying the newly proposed AeGVF snake to some real images, including 'human's cardiac CT image', 'airplane', 'cardiac MRI', 'monkey', 'human's lung CT image', 'hand', 'brain CT image', 'MRI of bladder', 'ultrasound image', and 'brain tumor'. The numerical results are depicted in Fig. 11 in yellow. The segmentation results are very impressive and all these show the promising of the proposed AeGVF snake.

5   Summary and conclusions

In this paper, we have proposed a new GVF model with advection enhancement, called AeGVF, for the active-contour image segmentation. This model is inspired by the functional derivative of an adaptive TV regularizer whose minimizer is expected to be able to preserve the desired object boundary effectively. The most distinguished feature of this newly proposed model is that it is equipped with an advection term $\nabla f \cdot \nabla V$ in solving the external force field V. To the authors' knowledge, such an advection effect has never been studied in the literature. With the advection enhancement to the gradient vectors,
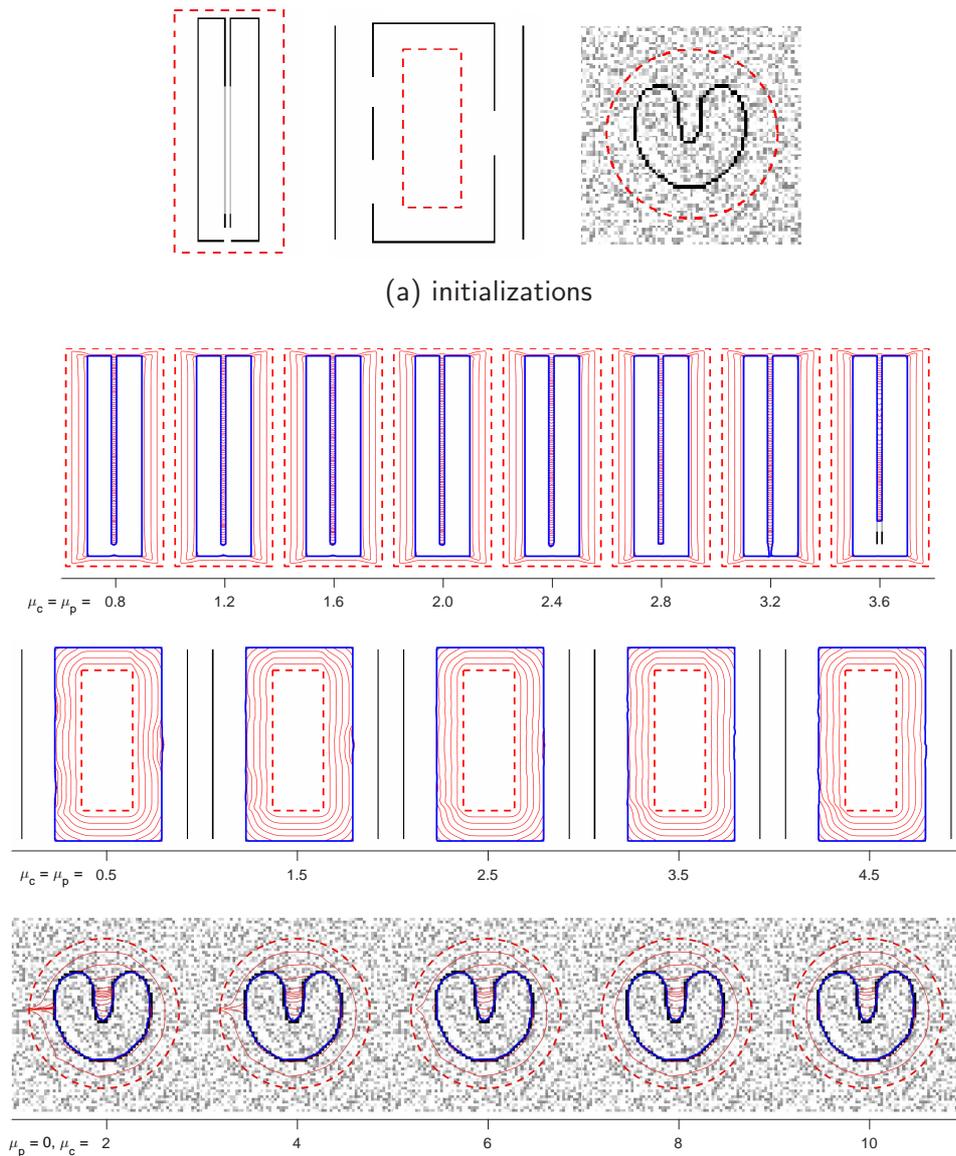
(a) initializations

Figure 10: Numerical results of the AeGVF model for Example 4.5. The top row illustrates the initializations for each tested image. In all cases, $\Delta t = 0.25$. From the second row to the bottom row, we set $(k,\sigma) = (0.01,0.15)$, $(0.01,0.01)$, and $(0,5,1)$, respectively. The blue curves depict the segmentation results of the AeGVF snake with different advection parameters.

the AeGVF snake is expected to be able to recover missing edges, to converge to a narrow and deep concavity, and to preserve weak edges very well. Indeed, the numerical results show that the AeGVF snake model is not only comparable to several improved GVF snake models such as GGVF, NGVF, NBGVF and ADF, it seems having much better segmentation quality than the others for most of the test images. More specifically, the
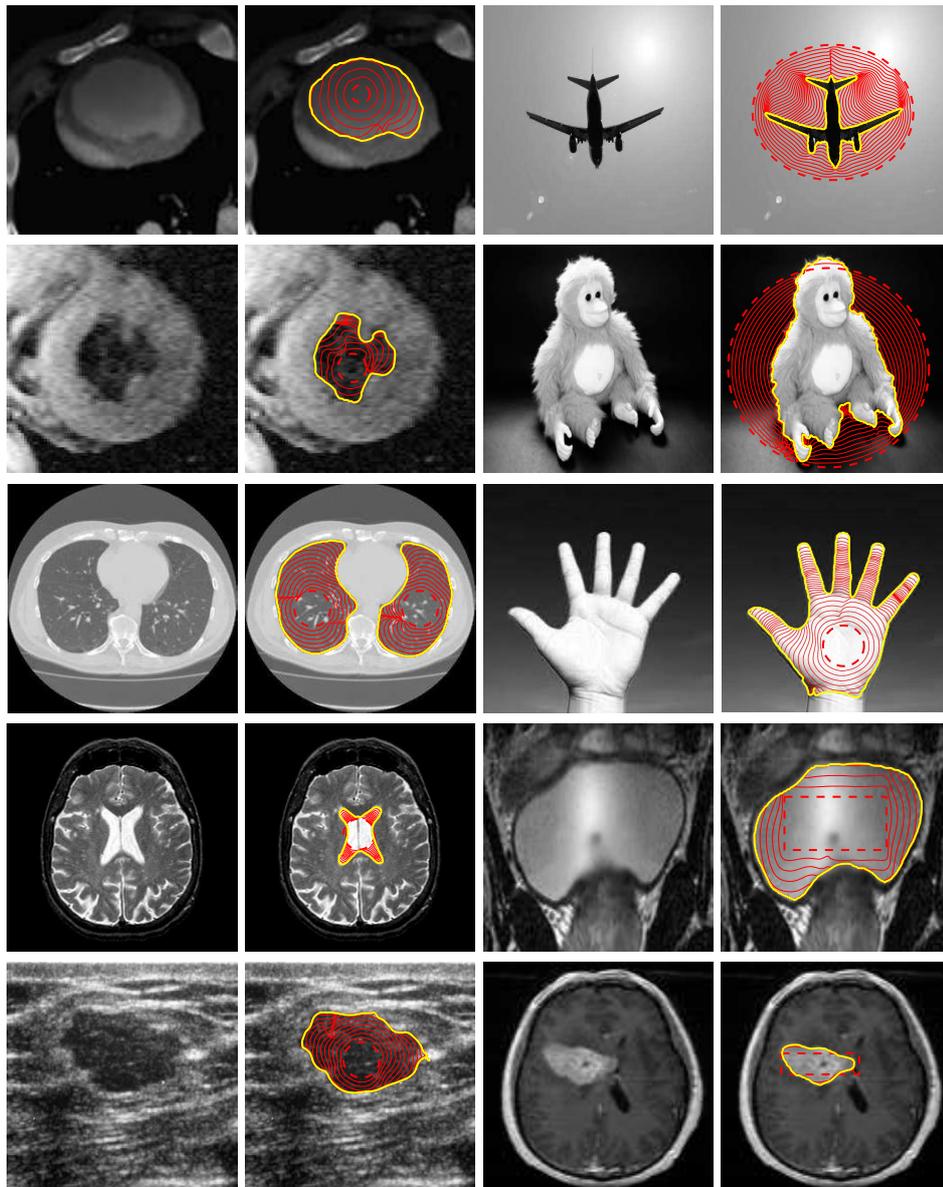
Figure 11: The segmentation results of the AeGVF snake for some real images.

AeGVF has the best performance and distinguished results in recovering missing edges and in converging to the LTIs with weak edges. Moreover, the AeGVF active-contour model performs very satisfactorily for several medical and natural images. Finally, we remark that a rigorous qualitative analysis of the newly proposed AeGVF model should be very interesting, and this deserves further study.

Acknowledgments

References

[1] G. Aronsson, M. G. Crandall, and P. Juutinen, A tour of the theory of absolutely minimizing functions, *Bulletin of the American Mathematical Society,* 41 (2004), pp. 439-505.

[2] G. Aubert and P. Kornprobst, *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, Springer-Verlag, New York, 2006.

[3] A. Azizi and K. Elkourd, Fast region-based active contour model driven by local signed pressure force, *Electronic Letters on Computer Vision and Image Analysis,* 15 (2016), pp. 1-13.

[4] X. Bresson, S. Esedoglu, P. Vandergheynst, J.-P. Thiran, and S. Osher, Fast global minimization of the active contour/snake model, *Journal of Mathematical Imaging and Vision,* 28 (2007), pp. 151-167.

[5] V. Caselles, F. Catte, T. Coll, and F. Dibos, A geometric model for active contours in image processing, *Numerische Mathematik,* 66 (1993), pp. 1-31.

[6] V. Caselles, R. Kimmel, and G. Sapiro, Geodesic active contours, *International Journal of Computer Vision,* 22 (1997), pp. 61-79.

[7] V. Caselles, J.-M. Morel, and C. Sbert, An axiomatic approach to image interpolation, *IEEE Transactions on Image Processing,* 7 (1998), pp. 376-386.

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

[9] L. D. Cohen and I. Cohen, Finite-element methods for active contour models and balloons for 2-D and 3-D images, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 15 (1993), pp. 1131-1147.

[10] L. D. Cohen and R. Kimmel, Global minimum for active contour models: A minimal path approach, *International Journal of Computer Vision*, 24 (1997), pp. 57-78.

[11] R. Durikovic, K. Kaneda, and H. Yamashita, Dynamic contour: A texture approach and contour operations, *The Visual Computer,* 11 (1995), pp. 277-289.

[12] P.-W. Hsieh, P.-C. Shao, and S.-Y. Yang, A regularization model with adaptive diffusivity for variational image denoising, *Signal Processing,* 149 (2018), pp. 214-228.

[13] T. Kajishima and K. Taira, *Computational Fluid Dynamics: Incompressible Turbulent Flows*, Springer-Verlag, Berlin, 2016.

[14] M. Kass, A. Witkin, and D. Terzopoulos, Snakes: active contour models, *International Journal of Computer Vision,* 1 (1988), pp. 321-331.

[15] A. Kovacs and T. Sziranyi, Harris function based active contour external force for image segmentation, *Pattern Recognition Letters,* 33 (2012), pp. 1180-1187.

[16] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady State and Time Dependent Problems*, SIAM, Philadelphia, 2007.

[17] F. Leymarie and M. D. Levine, Tracking deformable objects in the plane using an active contour model, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 15 (1993), pp. 617-634.

[18] B. Li and S. T. Acton, Active contour external force using vector field convolution for image segmentation, *IEEE Transactions on Image Processing,* 16 (2007), pp. 2096-2106.

[19] C. Li, J. Liu, and M. D. Fox, Segmentation of external force field for automatic initialization and splitting of snakes, *Pattern Recognition,* 38 (2005), pp. 1947-1960.

[20] R. Malladi, J. A. Sethian, and B. C. Vemuri, Shape modeling with front propagation: a level set approach, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 17 (1995), pp. 158-175.

[21] T. McInerney and D. Terzopoulos, A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4D image analysis, *Computerized Medical Imaging and Graphics,* 19 (1995), pp. 69-83.

[22] A. K. Mishra, P. W. Fieguth, and D. A. Clausi, Decoupled active contour (DAC) for boundary detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 33 (2011), pp. 310-324.

[23] J. Ning, C. Wu, S. Liu, and S. Yang, NGVF: an improved external force field for active contour model, *Pattern Recognition Letters,* 28 (2007), pp. 58-63.

[24] L. I. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D,* 60 (1992), pp. 259-268.

[25] M. Shemesh and O. Ben-Shahar, Free boundary conditions active contours with applications for vision, In: *International Symposium on Visual Computing (ISVC),* (2011), pp. 180-191.

[26] F. Y. Shih and K. Zhang, Locating object contours in complex background using improved snakes, *Computer Vision and Image Understanding,* 105 (2007), pp. 93-98.

[27] K. Siddiqi, Y. B. Lauziere, A. Tannenbaum, and S. W. Zucker, Area and length minimizing flows for shape segmentation, *IEEE Transactions on Image Processing,* 7 (1998), pp. 433-443.

[28] D. Terzopoulos and K. Fleischer, Deformable models, *The Visual Computer,* 4 (1988), pp. 306-331.

[29] Y. Wang, L. Liu, H. Zhang, Z. Cao, and S. Lu, Image segmentation using active contours with normally biased GVF external force, *IEEE Signal Process Letters,* 17 (2010), pp. 875-878.

[30] Y. Wu, Y. Wang, and Y. Jia, Adaptive diffusion flow active contours for image segmentation, *Computer Vision and Image Understanding,* 117 (2013), pp. 1421-1435.

[31] X. Xie and M. Mirmehdi, RAGS: region-aided geometric snake, *IEEE Transactions on Image Processing,* 13 (2004), pp. 640-652.

[32] C. Xu and J. L. Prince, Snakes, shapes, and gradient vector flow, *IEEE Transactions on Image Processing,* 7 (1998), pp. 359-369.

[33] C. Xu and J. L. Prince, Generalized gradient vector flow external forces for active contours, *Signal Processing,* 71 (1998), pp. 131-139.

[34] S. Zhu and R. Gao, A novel generalized gradient vector flow snake model using minimal surface and component-normalized method for medical image segmentation, *Biomedical Signal Processing and Control,* 26 (2016), pp. 1-10.