Minimizing Discrete Total Curvature for Image Processing

Qiuxiang Zhong, Yutong Li, Yijie Yang, Yuping Duan*
Center for Applied Mathematics, Tianjin University, China, Tianjin, 30070
{zhongqiuxiang, yutong_li, yangyijie, yuping.duan}@tju.edu.cn

Abstract

The curvature regularities have received growing attention with the advantage of providing strong priors in the continuity of edges in image processing applications. However, owing to the non-convex and non-smooth properties of the high-order regularizer, the numerical solution becomes challenging in real-time tasks. In this paper, we propose a novel curvature regularity, the total curvature (TC), by minimizing the normal curvatures along different directions. We estimate the normal curvatures discretely in the local neighborhood according to differential geometry theory. The resulting curvature regularity can be regarded as a re-weighted total variation (TV) minimization problem, which can be efficiently solved by the alternating direction method of multipliers (ADMM) based algorithm. By comparing with TV and Euler's elastica energy, we demonstrate the effectiveness and superiority of the total curvature regularity for various image processing applications.

1. Introduction

Curves and surfaces are important geometric elements in image analysis. As basic measurements of curves, both length and curvatures are natural regularities, which have been widely used in various image processing problems [15, 12, 13, 10]. For example, curvature is a natural regularizer for thin structures [8] and can preserve contrast and corners for image denoising [22].

Let $u : \Omega \to \mathbb{R}$ be an image defined on a domain $\Omega \subset \mathbb{R}^2$ with Lipschitz continuous boundary, and $u(x)$ denote the intensity of the gray level at point $x \in \Omega$. Assume that u is smooth, then the level set

$$\gamma_\lambda = \{x \in \Omega \mid u(x) = \lambda\}$$

is a smooth 1D manifold. Although images are rarely smooth functions, we can assume they are in certain functional space, e.g., $L^2(\Omega)$. Then the point-wise values are not so crucial as they are a set of zero measure. Suppose that the curve length is used as the geometric measurement, we then have

$$E[\gamma_\lambda] = \int_{\gamma_\lambda} ds.$$

When integrating all level curves of an image u, it gives exactly the Rudin-Osher-Fatemi TV model [14]

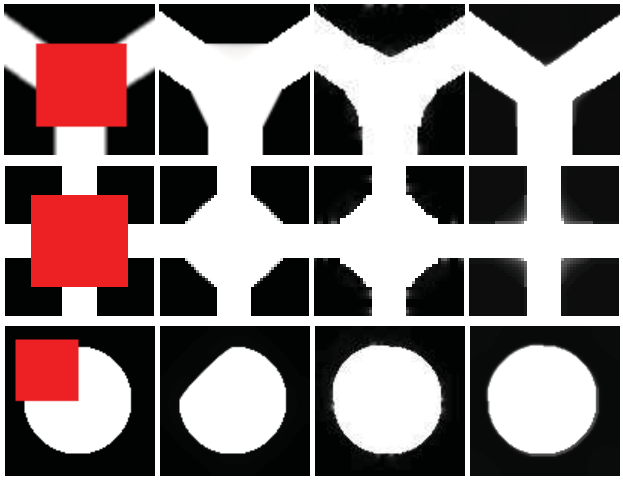
$$E[u] = \int_{-\infty}^{+\infty} \int_{\gamma_\lambda} |\nabla u| d\sigma ds = \int_\Omega |\nabla u| dx,$$

with ds and $d\sigma$ denoting the arclength of the level sets and their gradient flows. Considering the curvature curve model, the resulting image model becomes

$$E[u] = \int_\Omega \phi(\kappa) |\nabla u| dx, \tag{1}$$

where $\phi(\kappa)$ is some suitable functions of the curvature, e.g.,

$$\phi(\kappa) = \begin{cases} 1 + \alpha|\kappa|, & \text{Total Absolute Curvature,} \\ \sqrt{1 + \alpha|\kappa|^2}, & \text{Total Roto-translation Variation,} \\ 1 + \alpha|\kappa|^2, & \text{Total Squared Curvature,} \end{cases}$$



(a) Mask (b) TV (c) Euler's elastica (d) Total curvature

Figure 1. Total curvature is an ideal regularity for image inpainting, where the red regions represent inpainting domains. The parameters in our model (9) are set as $\lambda = 5$ and $\alpha = 0.5$.

*Corresponding author. The work was supported by NSFC 11701418, Major Science and Technology Project of Tianjin 18ZXRHSY00160 and Recruitment Program of Global Young Expert.

with α being a positive constant. For a 2D curve, the curvature of the level lines is defined as

$$\kappa = \kappa(x) := \nabla \cdot \Big(\frac{\nabla u}{|\nabla u|}\Big).$$

Such curvature energy was originally studied by Euler in modeling the shape of a torsion-free thin rod, and was introduced into computer vision by Mumford in [11]. Since then, numerous studies have been carried out to introduce Euler's elastica into various image processing tasks. For example, Masnou and Morel [9] proposed the regularity energy with $\phi(\kappa) = 1 + |\kappa|^p$ $(p \geq 1)$ for disoccluding noiseless images. Euler's elastica energy allows for the automatic handling of topologically complex inpainting domains, which can overcome the shortage of TV regularization in inpainting of incomplete images with large-scale missing domains. Figure 1 shows three images of simple shapes with their inpainting domains and the solutions of minimizing TV, Euler's elastica energy and our curvature regularity, respectively. From perceptual experiments, it is known that human visual system tends to complete partially occluded boundaries following a principle of good continuation. As can be seen, the Euler's elastica model gives obvious better restoration results than TV model, such that TV gives the shortest connection between the two T-junction points by straight lines and Euler's elastica tends to connect the T-junctions points with smoother edges. Due to the great success of Euler's elastica in image inpainting, it has been applied to other image processing applications such as denoising [18], segmentation [24], illusory contour [7] and segmentation with depth [23].

However, because of the nonquadratic high-order elastica term, the numerical minimization of the Euler's elastica-based energies are very challenging. The first numerical attempt to directly solve the Euler's elastica model was given by Shen, Kang and Chan in [17] based on the calculus of variation and the steepest descent method. Fast algorithms have been developed based on augmented Lagrangian method in [18, 19, 3]. Approximated convex envelope of the elastica energy was proposed and solved by the primal-dual algorithm in [6]. The curvature depending variational energies including Euler's elastica were presented using a convex functional in the roto-translation space and solved by the primal-dual scheme in [2].

Other curvature-based models include minimizing the mean curvature [22] and Gaussian curvature [20] for image processing tasks. Such curvature regularities can well preserve the geometric properties such as keeping corners of objects and grayscale intensity contrasts of images. However, similar to Euler's elastica energies, either high-order Partial Differential Equations (PDEs) are required to deal with or splitting strategies are employed to solve the high-order nonlinear minimization problems. Goldluecke and



(a) Mask image (b) Euler's elastica (c) Total curvature

Figure 2. Total curvature regularity works well on image segmentation of elongated structures. The parameters in our model (9) are set as $\lambda = 200$ and $\alpha = 0.1$.

Cremers [4] introduced the Menger-Melnikov curvature of the Radon measure, which was reformulated as weighted TV minimization with the weight function containing curvature information estimated from the observed image data. In [5, 21], efficient curvature filters were developed using the pixel-local analytical solutions to approximate the mean curvature and Gaussian curvature by enumerating the linear and developable surfaces in a 3×3 pixel neighborhood. However, the curvatures were roughly approximated by certain distances without rigorous definitions.

In this work, we introduce the *total curvature* regularity, which minimizes curvature functions of all the normal curvatures. In contrast to the classical curvature-based variational models, such as Euler's elastica and mean curvature models requiring the images at least twice differentiable, our approach estimates the normal curvatures in the discrete setting. By estimating the curvature separately, we can regard the curvature energy as a re-weighted TV minimization, where the weight function is updated iteratively to capture the curvature information accurately. As a consequence, the efficient and stable ADMM-based numerical scheme can be applied to solve the proposed total curvature regularized minimization problem. Compared with state-of-the-art variational models, the proposed method has three advantages as follows.

1) We introduce the concept of total curvature, which is the ℓ^1 norm of the normal curvatures along different directions over each point of the isolines. The novel curvature regularity can achieve better results than Euler's elastica for different image processing tasks. As shown, minimizing the total normal curvature can ideally recover the corners and edges of images with large-scale missing domains (see Figure 1) and accurately segment the objects with elongated structures (see Figure 2).

2) The normal curvatures are estimated point-wise in the discrete setting without requiring to solve any high-order PDE. Compared to the ADMM algorithm for the Euler's elastica model in [18], our ADMM based algorithm introduces only one artificial variable (three artificial variables in [18]), such that less computational costs are required in each iteration. The resulting numerical algorithm is of high efficiency, which saves

half of the CPU time consumed by the Euler's elastica model for image inpainting problems.

3) Our model is flexible to adapt with different function-types of curvature without affecting the way of the operator splitting and the associated ADMM-based algorithm.

We implement the total curvature regularity on image denoising, segmentation and inpainting problems. By comparing with the TV and Euler's elastica based approaches, the results demonstrate that the total curvature is a suitable regularizer for different image processing tasks.

2. Discrete Total Curvature

2.1. Normal curvatures

Let $\mathcal{S} : \boldsymbol{r} = \boldsymbol{r}(x, y)$ be a regular surface in $\mathbb{R}^3$, $O \in \mathcal{S}$ be a point and $\boldsymbol{t} \in T_O$ be a tangent vector with T_O denoting the tangent plane at point O. The normal curvature of $\mathcal{S}$ at O in the direction $\boldsymbol{t}$ characterizes the derivation of the surface at point O along the direction $\boldsymbol{t}$ from its tangent plane. Simultaneously, the normal curvature at O in direction $\boldsymbol{t}$ is also the normal curvature of arclength parameterized curve $\gamma(s) \in \mathcal{S}$ with $\gamma(0) = O$ and $\gamma'(0) = \boldsymbol{t}$, which is independent of the choice of γ. The normal curvature can be expressed as the amount of the curvature of γ in the direction of the surface normal $\boldsymbol{N}$

$$\kappa_n = \gamma'' \cdot \boldsymbol{N} = \frac{\mathrm{II}}{\mathrm{I}}, \tag{2}$$

which can be also expressed as the quotient of the second fundamental form II and the first fundamental form I of the surface. Thus, the normal curvature corresponding to direction $\boldsymbol{t}$ can be estimated using I and II according to (2) over each point on the surface. It is well-known that the first fundamental form can be approximated using the arclength, while there is the following result for the second fundamental form.

Proposition 2.1. *Suppose $\mathcal{S}$: $\boldsymbol{r} = \boldsymbol{r}(x, y)$ is a regular parametric surface, $O(x_0, y_0)$ is an arbitrary point on S, then the second fundamental form at point O can be estimated by*

$$\mathrm{II} \approx 2d, \tag{3}$$

where d denotes the projection distance of its neighboring point $H(x_0 + \Delta x, y_0 + \Delta y)$ to the tangent plane of O.

Proof. The projection distance of point $H(x_0 + \Delta x, y_0 + \Delta y)$ to the tangent plane can be defined as follows

$$d(\Delta x, \Delta y) = (\boldsymbol{r}(x_0 + \Delta x, y_0 + \Delta y) - \boldsymbol{r}(x_0, y_0)) \cdot \boldsymbol{N}.$$

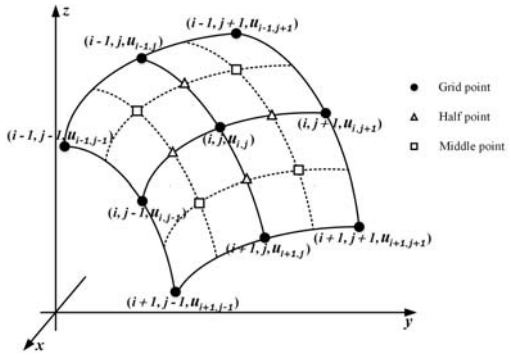


Figure 3. The 3-D grid mesh.

By Taylor's formula we have

$$\begin{aligned}&\boldsymbol{r}(x_0 + \Delta x, y_0 + \Delta y) - \boldsymbol{r}(x_0, y_0)\\ &= (\boldsymbol{r}_x \Delta x + \boldsymbol{r}_y \Delta y) + \frac{1}{2}[\boldsymbol{r}_{xx}(\Delta x)^2 + 2\boldsymbol{r}_{xy}\Delta x \Delta y\\ &\quad + \boldsymbol{r}_{yy}(\Delta y)^2] + \boldsymbol{o}((\Delta x)^2 + (\Delta y)^2),\end{aligned}$$

and

$$\lim_{(\Delta x)^2 + (\Delta y)^2 \to 0} \frac{\boldsymbol{o}((\Delta x)^2 + (\Delta y)^2)}{(\Delta x)^2 + (\Delta y)^2} = 0.$$

Owing to $\boldsymbol{r}_x \cdot \boldsymbol{N} = \boldsymbol{r}_y \cdot \boldsymbol{N} = 0$, it follows that

$$\begin{aligned}d(\Delta x, \Delta y) =& \frac{1}{2}[L(\Delta x)^2 + 2M\Delta x \Delta y + N(\Delta y)^2]\\ &+ o((\Delta x)^2 + (\Delta y)^2),\end{aligned}$$

where $L = \boldsymbol{r}_{xx} \cdot \boldsymbol{N}$, $M = \boldsymbol{r}_{xy} \cdot \boldsymbol{N}$, $N = \boldsymbol{r}_{yy} \cdot \boldsymbol{N}$, and $L(\Delta x)^2 + 2M\Delta x \Delta y + N(\Delta y)^2$ is the exact *second fundamental form*. Thus, when $\sqrt{(\Delta x)^2 + (\Delta y)^2} \to 0$, we have

$$\mathrm{II} \approx 2d(\Delta x, \Delta y),$$

which completes the proof. □

2.2. Calculation of normal curvatures

By considering the *image surface* or *graph* in $\mathbb{R}^3$ characterized with $z = u_{i,j}$, $(i, j) \in \Omega$, the image minimization problems are then transferred to the corresponding surface minimization problems. Without loss of generality, we represent a gray image as a $m \times n$ matrix and the grid $\Omega = \{(i, j) : 1 \le i \le m, 1 \le j \le n\}$. Similar to Euler's elastica energies, we use the staggered grid in the $x-y$ plane. As shown in Figure 3, the •-nodes, △-nodes and □-nodes represent the original points, half points and middle points, respectively. The intensity values on △-nodes are estimated as the mean of its two neighboring •-nodes, while on □-nodes are estimated as the mean of the four surrounding •-nodes.

The tangent planes can be defined by either the triangles or rectangles, where we use the triangle representation for the ease of computation. Then, every tangent plane defines

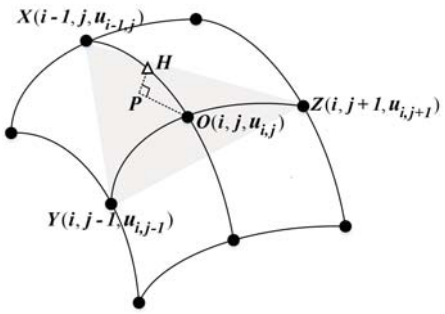


Figure 4. Calculate the normal curvature on image surface, the gray triangle in which represents a tangent plane of center point O.

a normal curvature. Figure 4 illustrates an example of the tangent plane at point O, which is denoted as T_{XYZ}. Given the 3D coordinates of X, Y and Z, the normal vector $\boldsymbol{N}$ of T_{XYZ} can be decided by the cross product of the vector $\overrightarrow{XY}$ and $\overrightarrow{XZ}$ as follows

$$\boldsymbol{N} = \overrightarrow{XY} \times \overrightarrow{XZ} = (2u_{i,j} - u_{i,j-1} - u_{i,j+1}, u_{i,j-1} - u_{i,j+1}, 2).$$

According to Proposition 2.1, we use the half point $H(i - \frac{1}{2}, j, u_{i-\frac{1}{2},j})$ to estimate the projection distance d to the tangent plane T_{XYZ} as

$$\begin{aligned} d &= \overrightarrow{PH} \cdot \boldsymbol{N} \qquad (4) \\ &= \frac{2u_{i,j} - u_{i,j-1} - u_{i,j+1}}{2\sqrt{(2u_{i-1,j} - u_{i,j-1} - u_{i,j+1})^2 + (u_{i,j-1} - u_{i,j+1})^2 + 4}}. \end{aligned}$$

On the other hand, the arclength $\widehat{OH}$ can be approximated in the following way

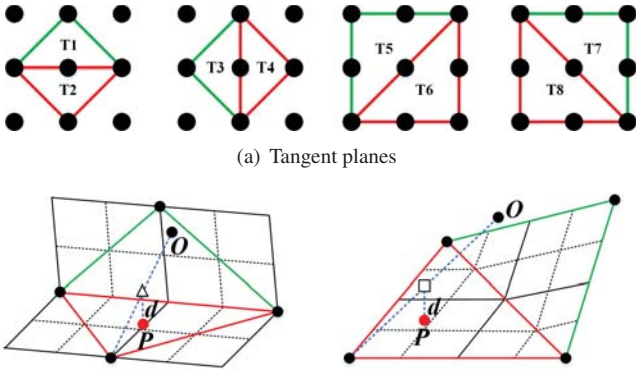
$$ds = \widehat{OH} \approx \sqrt{(u_{i-\frac{1}{2},j} - u_{i,j})^2 + h^2}, \qquad (5)$$

where $u_{i-\frac{1}{2},j}$ can be estimated as the mean of its two neighboring points $u_{i-1,j}$ and $u_{i,j}$, and h is grid size fixed as the same in both x-axis and the y-axis. Therefore, the normal curvature of point O in direction $\overrightarrow{OH}$ can be expressed as

$$\kappa_n \approx \frac{2d}{ds^2}. \qquad (6)$$

2.3. Discrete total curvature

From surface perspective, when the directional vector changes, there are many (perhaps infinitely many) normal curvatures. In our formulation, we estimate the point-wise normal curvatures in a 3×3 local window. Figure 5 displays the eight triangle planes (i.e., T1-T8) located physically nearest to the center point O, which are used to approximate the tangent planes in different directions. Note that the eight directions are chosen pairwise centrosymmetric to avoid the grid bias. The tangent planes can be divided into two categories such that T1-T4 use the half points and T5-T8 use the middle points to compute the projections. One example



(a) Tangent planes

(b) T2 (c) T8

Figure 5. The eight tangent planes of the center point in a 3×3 local window and the representatives of the two different kinds of projections.

of each type (i.e., T2 and T8) are also illustrated in Figure 5, where $\triangle$ and $\square$ points are used to compute the projections, respectively.

Given the pairwise neighboring point and tangent plane, we can compute the discrete normal curvatures accordingly. More precisely speaking, we calculate the distances d_ℓ, $\ell = 1, \ldots, 8$, of $(i, j, u_{i,j})$ corresponding to the eight tangent planes using (4), which are given as

$$\begin{aligned}
d_1 &= \frac{2u_{i,j} - u_{i,j-1} - u_{i,j+1}}{2\sqrt{(2u_{i-1,j} - u_{i,j-1} - u_{i,j+1})^2 + (u_{i,j-1} - u_{i,j+1})^2 + 4}}; \\
d_2 &= \frac{u_{i,j-1} + u_{i,j+1} - 2u_{i,j}}{2\sqrt{(2u_{i+1,j} - u_{i,j-1} - u_{i,j+1})^2 + (u_{i,j+1} - u_{i,j-1})^2 + 4}}; \\
d_3 &= \frac{u_{i-1,j} + u_{i+1,j} - 2u_{i,j}}{2\sqrt{(u_{i+1,j} - u_{i-1,j})^2 + (u_{i-1,j} + u_{i+1,j} - 2u_{i,j-1})^2 + 4}}; \\
d_4 &= \frac{2u_{i,j} - u_{i-1,j} - u_{i+1,j}}{2\sqrt{(u_{i-1,j} - u_{i+1,j})^2 + (u_{i-1,j} + u_{i+1,j} - 2u_{i,j+1})^2 + 4}}; \\
d_5 &= \frac{u_{i-1,j-1} + u_{i-1,j+1} + u_{i+1,j-1} - u_{i,j} - u_{i-1,j} - u_{i,j-1}}{2\sqrt{(u_{i+1,j-1} - u_{i-1,j-1})^2 + (u_{i-1,j+1} - u_{i-1,j-1})^2 + 4}}; \\
d_6 &= \frac{u_{i,j} + u_{i+1,j} + u_{i,j+1} - u_{i+1,j+1} - u_{i-1,j+1} - u_{i+1,j-1}}{2\sqrt{(u_{i-1,j+1} - u_{i+1,j+1})^2 + (u_{i+1,j-1} - u_{i+1,j+1})^2 + 4}}; \\
d_7 &= \frac{u_{i,j} + u_{i-1,j} + u_{i,j+1} - u_{i-1,j-1} - u_{i+1,j+1} - u_{i-1,j+1}}{2\sqrt{(u_{i-1,j+1} - u_{i+1,j+1})^2 + (u_{i-1,j-1} - u_{i-1,j+1})^2 + 4}}; \\
d_8 &= \frac{u_{i-1,j-1} + u_{i+1,j+1} + u_{i+1,j-1} - u_{i,j} - u_{i+1,j} - u_{i,j-1}}{2\sqrt{(u_{i+1,j-1} - u_{i-1,j-1})^2 + (u_{i+1,j+1} - u_{i+1,j-1})^2 + 4}}.
\end{aligned}$$

Then, we compute the arclengths of the central point $(i, j, u_{i,j})$ to its neighboring points (i.e., half points or middle points), which are defined as the square root of the quadratic sum of the intensity differences and the grid size between two points according to (5). As a result, the eight normal curvatures can be calculated using (6) over each point, which gives

$$\kappa_\ell \approx \begin{cases} \frac{2d_\ell}{(u_\ell^{\triangle} - u_{i,j})^2 + h^2}, & \ell = 1,\ 2,\ 3,\ 4, \\ \frac{2d_\ell}{(u_\ell^{\square} - u_{i,j})^2 + 2h^2}, & \ell = 5,\ 6,\ 7,\ 8, \end{cases} \qquad (7)$$

with $u_\ell^{\triangle}$ and $u_\ell^{\square}$ being the intensity of the half points and middle points as shown in Figure 3.

It is well-known the maximum and minimum curvature (i.e. principal curvatures) among all the normal curvatures can completely determine all the normal curvatures in theory. However, only limited numbers of normal curvatures (i.e. eight in total) are enumerated over a local window in our formulation. Thus, we introduce the discrete total curvature as a geometric measurement of the image function over each point $x \in \Omega$, which minimizes the following ℓ^1 norm of all normal curvatures along different directions

$$\kappa(x) = \int_0^{2\pi} |\kappa_n(\theta)| d\theta \approx \sum_{\ell=1}^{8} |\kappa_\ell|. \tag{8}$$

3. Numerical Algorithm

With the discrete total curvature (8), we can formulate the curvature-based energy as the following minimization problem

$$\min_u \int_\Omega \phi(\kappa)|\nabla u| dx + \lambda \mathcal{D}(u, f). \tag{9}$$

Note that we regard the total curvature as curvature of the level sets in (9) for the ease of computation. Here, the data fidelity term $\mathcal{D}(\cdot,\cdot)$ measures the distance between the unknown u and the observed data f, which varies according to different image processing tasks such as

$$\mathcal{D}(u,f) = \begin{cases} \frac{1}{2}\|u-f\|^2, & \text{for denoising;} \\ \langle u, f_1 - f_2\rangle, & \text{for segmentation;} \\ \frac{1}{2}\|u-f\|^2_{\Omega\setminus X}, & \text{for inpainting;} \end{cases} \tag{10}$$

with $X \subset \Omega$ denoting domain to be inpainted for inpaiting problem and f_1, f_2 being nonnegative potential functions for segmentation problem.

We introduce an auxiliary variable $\boldsymbol{v}$ and rewrite the original unconstrained optimization problem (9) into the following equivalent constrained minimization

$$\begin{aligned} &\min_{u,\boldsymbol{v}} \int_\Omega \phi(\kappa)|\boldsymbol{v}| dx + \lambda \mathcal{D}(u,f) \\ &\text{s.t. } \boldsymbol{v} = \nabla u. \end{aligned} \tag{11}$$

The associated augmented Lagrangian functional can be defined as follows

$$\begin{aligned} \mathcal{L}(u, \boldsymbol{v}; \boldsymbol{\Lambda}) = &\int_\Omega \phi(\kappa)|\boldsymbol{v}| dx + \lambda \mathcal{D}(u,f) \\ &+ < \boldsymbol{\Lambda}, \boldsymbol{v} - \nabla u > + \frac{\mu}{2}\|\boldsymbol{v} - \nabla u\|^2, \end{aligned}$$

where $\boldsymbol{\Lambda}$ represents the Lagrange multiplier, and μ is a positive penalty parameter. We iteratively and alternatively solve the u- and $\boldsymbol{v}$-subproblem until reaching the terminating condition. The ADMM-based algorithm is described in Algorithm 1.

Algorithm 1: ADMM for Model (9)

1: **Input:** Given image f, model parameters λ, α and μ, maximum iteration T_{max}, and stopping threshold ϵ.
2: **Initialize:** $u^0 = f$, $\boldsymbol{v}^0 = 0$, $\boldsymbol{\Lambda}^0 = 0$.
3: **while** (not converged and $k \le T_{max}$) do

(i) Compute u^{k+1} from:

$$\begin{aligned} u^{k+1} = \arg\min_u \Big\{ &\lambda \mathcal{D}(u,f) \\ &- < \boldsymbol{\Lambda}^k, \nabla u - \boldsymbol{v}^k > + \frac{\mu}{2}\big\|\nabla u - \boldsymbol{v}^k\big\|^2 \Big\}; \end{aligned} \tag{12}$$

(ii) Compute κ according to (8) using the latest estimation u^{k+1} and take it into $\phi(\kappa)$;

(iii) Compute $\boldsymbol{v}^{k+1}$ from:

$$\begin{aligned} \boldsymbol{v}^{k+1} = \arg\min_{\boldsymbol{v}} \Big\{ &\int_\Omega \phi(\kappa)|\boldsymbol{v}| dx \\ &+ < \boldsymbol{\Lambda}^k, \boldsymbol{v} - \nabla u^{k+1} > + \frac{\mu}{2}\big\|\boldsymbol{v} - \nabla u^{k+1}\big\|^2 \Big\}; \end{aligned} \tag{13}$$

(iv) Update $\boldsymbol{\Lambda}^{k+1}$ from:

$$\boldsymbol{\Lambda}^{k+1} = \boldsymbol{\Lambda}^k + \mu(\boldsymbol{v}^{k+1} - \nabla u^{k+1}); \tag{14}$$

(v) Check convergence condition:

$$\|u^{k+1} - u^k\|_1 \le \epsilon \|u^k\|_1.$$

4: **end while**

3.1. Sub-minimization w.r.t. u

Since all data fidelity terms in (10) are smooth, the first-order optimality condition of (12) can be expressed as the following linear equation

$$\lambda \nabla \mathcal{D}(u^{k+1}, f) + \mu \nabla^*(\nabla u^{k+1} - \boldsymbol{v}^k - \frac{\boldsymbol{\Lambda}^k}{\mu}) = 0, \tag{15}$$

with

$$\nabla \mathcal{D}(u,f) = \begin{cases} u - f, & \text{for denoising;} \\ f_1 - f_2, & \text{for segmentation;} \\ u_{\Omega\setminus X} - f_{\Omega\setminus X}, & \text{for inpainting.} \end{cases}$$

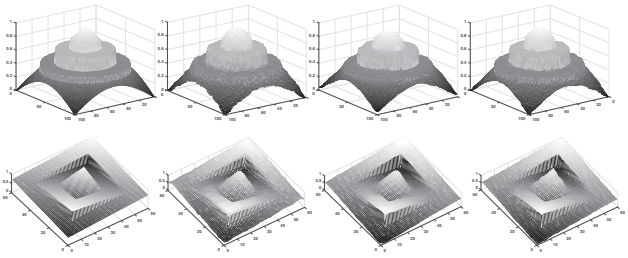
Thus, the solution of the u-subproblem can be denoted as

$$u^{k+1} = \begin{cases} \big(\lambda f + \nabla^*(\mu \boldsymbol{v}^k + \boldsymbol{\Lambda}^k)\big)/(\lambda \mathcal{I} - \mu\Delta); \\ \big(f_2 - f_1 + \nabla^*(\mu \boldsymbol{v}^k + \boldsymbol{\Lambda}^k)\big)/(\lambda \mathcal{I} - \mu\Delta); \\ \big(\lambda_X(x) f + \nabla^*(\mu \boldsymbol{v}^k + \boldsymbol{\Lambda}^k)\big)/(\lambda_X(x) - \mu\Delta); \end{cases}$$

where $\mathcal{I}$ denotes the identity matrix, $\lambda_X(x) = \lambda \cdot 1_{\Omega\setminus X}(x)$ with $1_{\Omega\setminus X} : \Omega \to \{0,1\}$ being the characteristic function of the region outside the inpainting domain. The above linear systems can be efficiently solved by either Fast Fourier Transform (FFTs) or iterative scheme such as Gauss-Seidel method.

Table 1. The PSNR of noise removal on grayscale and color images for TV, Euler's elastica (EE) and our model using total absolute curvature (TAC), total roto-translational variation (TRV) and total squared curvature (TSC), respectively.

Noise level	$\sigma = 20$					$\sigma = 30$					$\sigma = 40$				
Images	TV	EE	TAC	TRV	TSC	TV	EE	TAC	TRV	TSC	TV	EE	TAC	TRV	TSC
Boat	27.82	28.52	**29.09**	28.95	28.72	26.24	26.80	**27.22**	27.19	27.01	25.04	25.66	26.15	**26.18**	25.92
Cameraman	27.50	28.21	**29.07**	29.04	28.88	26.05	26.64	**27.13**	27.11	26.88	24.21	25.15	**25.81**	25.63	25.62
Couple	27.15	28.06	**28.55**	28.49	28.24	25.92	26.35	**26.78**	26.76	26.58	24.81	25.30	**25.69**	25.58	25.43
Hill	27.92	28.64	**29.17**	29.12	28.86	26.45	27.09	**27.50**	27.47	27.29	25.84	26.26	**26.64**	26.61	26.46
House	29.62	30.34	30.95	**30.98**	30.68	27.91	28.54	**29.06**	29.03	28.84	26.73	27.52	**28.13**	27.89	27.76
Lena	29.17	30.07	30.68	**30.72**	30.42	28.00	28.46	28.81	**28.83**	28.62	27.02	27.41	**27.88**	27.84	27.63
Man	28.19	28.71	**29.25**	29.23	29.03	26.70	27.18	**27.55**	27.50	27.36	25.77	26.16	26.57	**26.60**	26.37
Montage	28.90	29.80	**30.40**	30.30	30.14	26.63	27.47	**27.93**	27.85	27.76	24.58	25.55	**26.23**	26.08	25.98
Peppers	27.54	28.36	**29.06**	28.94	28.79	25.52	26.24	**26.84**	26.72	26.65	24.46	24.91	**25.38**	25.30	25.16
Mean	28.20	28.97	**29.58**	29.53	29.31	26.60	27.20	**27.64**	27.61	27.44	25.38	25.99	**26.50**	26.41	26.26
HouseRGB	29.19	29.87	**30.52**	30.48	30.31	27.35	28.03	**28.67**	28.55	28.41	26.38	27.04	**27.54**	27.47	27.33
LenaRGB	29.05	29.81	**30.42**	30.31	30.23	27.63	28.17	28.68	**28.72**	28.50	26.81	27.35	**27.80**	27.76	27.61
PeppersRGB	28.71	29.68	**30.18**	30.15	29.98	27.23	28.05	**28.54**	28.46	28.33	26.42	27.03	27.50	**27.53**	27.35
PlaneRGB	29.41	30.36	**30.88**	30.77	30.68	27.54	28.16	**28.77**	28.73	28.57	26.58	27.00	**27.60**	27.57	27.45
Mean	29.09	29.93	**30.50**	30.43	30.30	27.44	28.10	**28.67**	28.61	28.46	26.55	27.11	**27.61**	27.58	27.44



(a) Clean surface (b) TV (c) Euler's elastica (d) TC

Figure 6. The image surfaces of the clean images, denoising images obtained by TV, Euler's elastica and our TC model. The PSNR and SSIM of recovery images for different models: First row: (b) 35.45/0.9404; (c) 36.52/0.9515; (d) **37.81/0.9635**; Second row: (b) 33.84/0.9216; (c) 35.02/0.9484; (d) **36.06/0.9548**.

3.2. Sub-minimization w.r.t. v

The minimization problem w.r.t. $\boldsymbol{v}$ becomes straightforward, which has the unique minimizer using the shrinkage operator [1]

$$\boldsymbol{v}^{k+1} = \text{shrinkage}\left(\nabla u^{k+1} - \frac{\boldsymbol{\Lambda}^k}{\mu}, \frac{\phi(\kappa(u^{k+1}))}{\mu}\right) \quad (16)$$

with the shrinkage operator being defined as

$$\text{shrinkage}(a, b) = \max\{|a| - b, 0\} \circ \frac{a}{|a|},$$

and $\circ$ being the element-wise multiplication.

4. Numerical Experiments

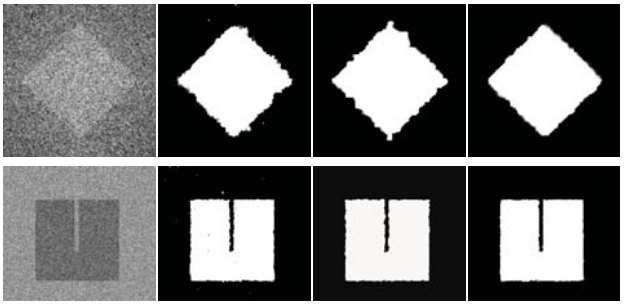
In this section, comprehensive experiments on image denoising, segmentation and inpainting are conducted to verify the efficiency and superiority of our total curvature regularity. The experimental images are composed of different edges and texture structures as well as homogenous regions. To setup the experimental comparison as fair as possible, the parameters of the comparative methods are selected as suggested in the corresponding papers. All numerical experiments are performed utilizing Matlab R2016a on a machine with 3.40GHz Intel(R) Core(TM) i7-6700 CPU and 32GB RAM.

4.1. Parameters discussing

There are three tunable parameters in our Algorithm 1, i.e., λ, α, μ. The most important one is the regularization parameter λ, which is used to balance the contribution between the data fidelity and curvature regularity. The smaller the λ is, the smoother the reconstruction is. The positive parameter α can balance the influence between the curvature and arclength, which should be chosen appropriately to smooth the homogenous regions as well as preserve the image details. The penalty parameter μ controls the convergent speed and stability of the proposed algorithm. We notice that large μ reduces both efficiency of the algorithm and restoration quality, while too small μ can not guarantee the stability of proposed algorithm. The specific values of λ, α and μ are provided in each experiment. Besides, we choose $h = 1$ throughout the experiments for the best balance between the smoothness and fine details.

4.2. Image denoising

In the first place, we evaluate the proposed curvature regularized model on two smooth images, which are degraded by Gaussian noise with zero mean and the standard deviation $\sigma = 10$. We compare the denoising performance with the TV-based model [14] and the Euler's elastica model [18]. The experience-dependent parameters for our approach are set as $\lambda = 1.0$, $\mu = 20$, $\alpha = 5$, $T_{max} = 300$ and $\epsilon = 2 \times 10^{-5}$. Figure 6 displays the image surfaces plotted using the clean images and the restored images from the TV, Euler's elastica and our TC model. It is shown that our approach preserves the jumps and sharp corners better than the Euler's elastica model, while gives much smoother



(a) Noisy images (b) TV (c) Euler's elastica (d) TC

Figure 7. Segmentation for noisy images. (a) Noisy image ('Dimond' [128 × 128] and 'Square' [228 × 202]); (b)-(d) segmentations of TV, Euler's elastica and our TC model, respectively. The parameters of the TC model are set as $\lambda = 1$, $\alpha = 0.1$.

Table 2. Evaluation results of noisy image segmentation for TV, Euler's elastica and TC model in Figure 7.

Images	Index	TV	Euler's Elastica	TC
Dimond	Precision	0.9732	0.9701	**0.9831**
	Recall	0.9804	0.9866	**0.9921**
	JS	0.9547	0.9575	**0.9755**
	CPU	**3.26**	5.54	3.30
Square	Precision	0.9954	0.9954	**0.9977**
	Recall	0.9860	0.9961	**0.9980**
	JS	0.9915	0.9916	**0.9957**
	CPU	**5.40**	13.49	9.40

results than the TV model.

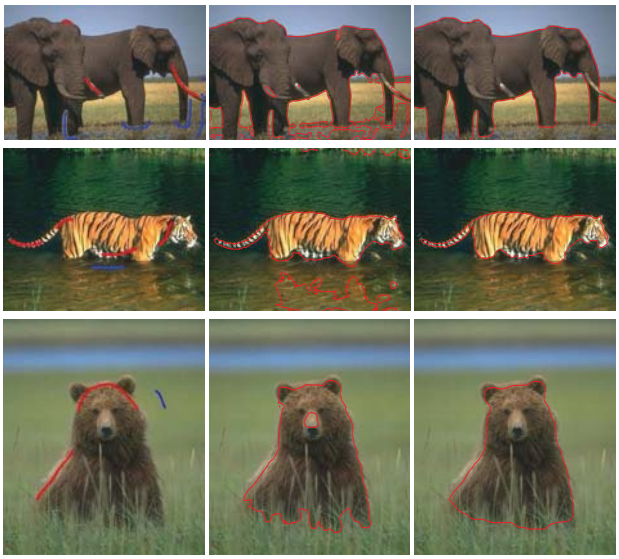
We further evaluate the denoising performance on both grayscale and color images, which are downloaded from the BM3D website[1]. Three types of curvature functions, i.e., TAC, TRV and TSC, are evaluated on the images corrupted by Gaussian noise with zero mean and the standard deviation $\sigma = \{20, 30, 40\}$. We fix the parameters $\mu = 30$, $T_{max} = 300$, $\epsilon = 5 \times 10^{-5}$, $\lambda = \{0.6, 0.4, 0.3\}$ for $\sigma = \{20, 30, 40\}$, respectively. And $\alpha = \{5, 30, 2.5\}$ are used in TAC, TRV and TSC models to balance the effect of curvature and arclength in minimizing the curvature energy. The algorithm parameters of TV and Euler's elastica models are well selected as recommended in their original papers. Table 1 details the values of PSNR obtained by our proposals and the competing methods, where our models always achieve higher PSNR than the other two models, especially the TAC gaining the overall 1dB and 0.5dB improvements of PSNR compared to the TV model and Euler's elastica model, respectively.

4.3. Image segmentation

We further apply our total curvature regularity to the figure-ground segmentation tasks, which aim to separate foreground objects from the background. The potential function f_1 and f_2 in (10) are determined by

$$f_1 = (f - c_1)^2 \quad \text{and} \quad f_2 = (f - c_2)^2,$$

[1] https://www.cs.tut.fi/~foi/GCF-BM3D/



(a) Mask images (b) Euler's elastica (c) TC

Figure 8. Supervised segmentation for natural images. (a) Mask images with pixels marked as foreground (red) and background (blue); (b) Segmentation of Euler's elastica; (c) Segmentation of our TC model. The parameters of the TC model are set as $\lambda = 1$, $\alpha = 0.1$.

where c_1 and c_2 denote the means of foreground and background, and are dynamically updated in the scheme. Figure 7 shows two experimental results of unsupervised segmentation, where both images are corrupted by serious white Gaussian white noises. Compared to TV and Euler's elastica energy [24], our TC model gives the results with visually smoother boundaries, which are also convinced by the values of precision, recall and Jaccard similarity (JS) in Table 2. Besides, much CPU time can be saved by explicitly estimating the curvatures in each iteration. For supervised image segmentation, we select three representative color images from the Berkeley segmentation dataset[2]. Figure 8 (b) and (c) display the segmentation results of Euler's elastica and our total curvature regularity using the same mask information as provided in Figure 8 (a), which demonstrate our total curvature model can provide more meaningful segmentation results with accurate boundaries for the elephants, tiger and bear, respectively.

4.4. Image inpainting

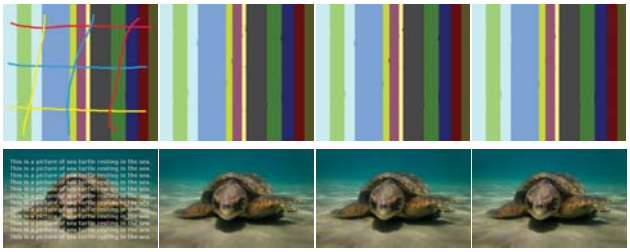
Last but not least, we demonstrate some examples of our TC model on image inpainting problems. Similar to Figure 1, we compare the results with both TV-based inpainting model [16] and Euler's elastica-based inpainting model [19] using the same termination conditions $T_{max} = 500$ and $\epsilon = 1 \times 10^{-4}$.

[2] https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/



(a) Mask images (b) TV (c) Euler's elastica (d) TC

Figure 10. The inpainting results of different real images ('Golf' $[268 \times 360]$ and 'Zoo' $[197 \times 313]$) by TV, Euler's elastica and our TC model. The parameters are set as: (b) TV: $\lambda = 5$; (c) Euler's elastica: $\lambda = 2 \times 10^3$ and $\alpha = 10$; (d) TC: $\lambda = 10$ and $\alpha = 20$.



(a) Mask images (b) TV (c) Euler's elastica (d) TC

Figure 9. The inpainting results of different color images ('Harmonic' $[315 \times 357]$ and 'Turtle' $[318 \times 500]$) by TV, Euler's elastica and our TC model. The parameters of the TC model are set as $\lambda = 5$, and $\alpha = 0.5$.

In Figure 9, we present two convincing examples obtained by the TV, Euler's elastica and our TC model on color image inpainting problems. The results of the TV inpainting present the piecewise constant values inside the inpainting domain, which are visually unnatural as displayed in Figure 9 (b). On the other hand, the Euler's elastica and our total curvature can connect the large gaps as well as protect image structures owing to the minimization of the curvature; see Figure 9 (c) and (d). Moreover, it is clearly shown that our TC model gives the visually best inpainting results, which well recovers the inpainting domain and preserves fine details. Also, Table 3 records the evaluations of the inpainting results in terms of PSNR, SSIM and CPU time, which verifies the efficiency and superiority of our total curvature model by providing significantly higher PSNR and SSIM and saving half of the CPU time consumed by the Euler's elastica model.

Moreover, we validate the performance of our TC model on real image inpainting applications. Our target is to fill in the red boxes labelled region in Figure 10 (a). The inpainting results of the TV, Euler's elastica and our TC model are displayed in Figure 10 (b), (c) and (d), respectively. The results evidence that the curvature regularities are important as otherwise the regions are filled in with mostly homogeneous intensity values. By comparing with the two curvature-based models, it is shown that our TC model gives better visual results, which well integrates with the surroundings. Besides, the computational time is significantly reduced using the discrete curvature regularization as demonstrated in Table 3.

Table 3. The PSNR, SSIM and CPU time of image inpainting experiments for TV, Euler's elastica and our TC model.

Images	Index	TV	Euler's Elastica	TC
Harmonic	PSNR	35.92	37.35	**38.73**
	SSIM	0.9826	0.9895	**0.9978**
	CPU	**12.49**	68.75	32.35
Turtle	PSNR	37.09	38.27	**39.65**
	SSIM	0.9724	0.9818	**0.9935**
	CPU	**40.38**	175.10	96.13
Golf	CPU	**11.63**	61.25	30.05
Zoo	CPU	**9.94**	42.66	20.96

5. Conclusion

In this work, we introduced the total curvature regularity for image processing applications, which can ideally preserve the geometric features by minimizing the ℓ^1 norm of the normal curvatures in different directions. Instead of solving any high-order PDE, we estimated the discrete normal curvatures in a local neighborhood, which can be computed efficiently relying on the differential geometry theory. Then, we solved the resulting curvature-based minimization problem using the ADMM algorithm, the two subproblems of which can be efficiently computed through the closed-form solutions. Numerical experiments on image denoising, segmentation and inpainting demonstrated the efficiency and effectiveness of the proposed curvature regularity.

References

[1] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[2] Antonin Chambolle and Thomas Pock. Total roto-translational variation. *Numerische Mathematik*, 142(3):611–666, 2019.

[3] Liang-Jian Deng, Roland Glowinski, and Xue-Cheng Tai. A new operator splitting method for the Euler elastica model for image smoothing. *SIAM Journal on Imaging Sciences*, 12(2):1190–1230, 2019.

[4] Bastian Goldluecke and Daniel Cremers. Introducing total curvature for image processing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1267–1274, 2011.

[5] Yuanhao Gong and Ivo F Sbalzarini. Curvature filters efficiently reduce certain variational energies. *IEEE Transactions on Image Processing*, 26(4):1786–1798, 2017.

[6] Stefan Heber, Rene Ranftl, and Thomas Pock. Approximate envelope minimization for curvature regularity. In *Proceedings of the Springer European Conference on Computer Vision*, pages 283–292, 2012.

[7] Sung Ha Kang, Wei Zhu, and Jackie Shen. Illusory shapes via corner fusion. *SIAM Journal on Imaging Sciences*, 7(4):1907–1936, 2014.

[8] Dmitrii Marin, Yuchen Zhong, Maria Drangova, and Yuri Boykov. Thin structure estimation with curvature regularization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 397–405, 2015.

[9] Simon Masnou and J-M Morel. Level lines based disocclusion. In *Proceedings of the IEEE International Conference on Image Processing*, pages 259–263, 1998.

[10] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness via curvature regularization, and vice versa. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9078–9086, 2019.

[11] David Mumford. Elastica and computer vision. In *Proceedings of the Springer Algebraic Geometry and Its Applications*, pages 491–506. 1994.

[12] Carl Olsson and Yuri Boykov. Curvature-based regularization for surface approximation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1576–1583, 2012.

[13] Carl Olsson, Johannes Ulén, Yuri Boykov, and Vladimir Kolmogorov. Partial enumeration and curvature regularization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2936–2943, 2013.

[14] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

[15] Thomas Schoenemann, Fredrik Kahl, and Daniel Cremers. Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 17–23, 2009.

[16] Jianhong Shen and Tony F Chan. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, 62(3):1019–1043, 2002.

[17] Jianhong Shen, Sung Ha Kang, and Tony F Chan. Euler's elastica and curvature-based inpainting. *SIAM Journal on Applied Mathematics*, 63(2):564–592, 2003.

[18] Xue-Cheng Tai, Jooyoung Hahn, and Ginmo Jason Chung. A fast algorithm for Euler's elastica model using augmented lagrangian method. *SIAM Journal on Imaging Sciences*, 4(1):313–344, 2011.

[19] Maryam Yashtini and Sung Ha Kang. A fast relaxed normal two split method and an effective weighted TV approach for Euler's elastica image inpainting. *SIAM Journal on Imaging Sciences*, 9(4):1552–1581, 2016.

[20] AR Yezzi. Modified curvature motion for image smoothing and enhancement. *IEEE Transactions on Image Processing*, 7(3):345–352, 1998.

[21] Hui Yin, Yuanhao Gong, and Guoping Qiu. Side window filtering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8758–8766, 2019.

[22] Wei Zhu and Tony Chan. Image denoising using mean curvature of image surface. *SIAM Journal on Imaging Sciences*, 5(1):1–32, 2012.

[23] Wei Zhu, Tony Chan, and Selim Esedoḡlu. Segmentation with depth: A level set approach. *SIAM Journal on Scientific Computing*, 28(5):1957–1973, 2006.

[24] Wei Zhu, Xue-Cheng Tai, and Tony Chan. Image segmentation using Euler's elastica as the regularization. *Journal of Scientific Computing*, 57(2):414–438, 2013.