Weighted mean curvature

Author(s):
Gong, Yuanhao; Goksel, Orcun (ID)

Weighted Mean Curvature

Yuanhao Gong and Orcun Goksel

Abstract—In image processing tasks, spatial priors are essential for robust computations, regularization, algorithmic design and Bayesian inference. In this paper, we introduce weighted mean curvature (WMC) as a novel image prior and present an efficient computation scheme for its discretization in practical image processing applications. We first demonstrate the favorable properties of WMC, such as sampling invariance, scale invariance, and contrast invariance with Gaussian noise model; and we show the relation of WMC to area regularization. We further propose an efficient computation scheme for discretized WMC, which is demonstrated herein to process over 33.2 giga-pixels/second on GPU. This scheme yields itself to a convolutional neural network representation. Finally, WMC is evaluated on synthetic and real images, showing its superiority quantitatively to total-variation and mean curvature.

I. INTRODUCTION

Recovering a signal from one or more observations is a fundamental task in image processing, such as in denoising, super-resolution, deconvolution, dehazing, and enhancement. The act of generating an observation from the physical space (or an original signal) is called the imaging process, while the model governing this process is the imaging model. Prior information on this model is a fundamental piece of assumption, which can determine the success or failure of image processing methods. A typical example is *image denoising*, where the observed data contains measurement errors or noise, which is commonly assumed to follow an expected distribution, given some prior knowledge on the imaging (observation) process. In *image smoothing*, the goal is to remove undesired details while preserving "major" structures, where the structure-detail differentiation is again based on some assumed priors.

Among priors, the well-known Total Variation (TV) assumes that the signal or image to be recovered is a piecewise-constant function [25], which has been used successfully in many image processing tasks over the years. Another potential prior is the assumption that the original image has minimal area [16], which enforces both the gradient and the normal changes to be small, hence resulting in smoother images. However, this minimal-area assumption is difficult to apply using conventional optimization algorithms. Alternatively, a general gradient distribution prior can be assumed, which imposes the gradient to satisfy certain distributions, rather than being minimized [10]. Beyond first-order information, higher-order quantities such as curvature can also be used as a prior. Popular choices are Gaussian curvature [13] and mean curvature [29], [5], [14], [11]. Since these higher-order priors already assume the original signal is higher-order differentiable, the resulting images are enforced to be smooth; consequently, often losing any sharp edge detail.

For an image processing task, even when the imaging model is the same, the use of different priors may lead to different results. For example, assuming TV will lead to piecewise-constant image results; while with area regularization, the final estimation will be close to a piecewise-minimal surface. Similarly, for Gaussian curvature regularization, the result would be close to a piecewise developable surface [14]. Accordingly, the prior and applied regularization may affect the results that can be expected from an image processing task in a major way, and a suitable prior is a crucial choice.

Prior choice may also be affected by the available or affordable solution strategy, since imposing some priors require specific numerical solvers for computation. For instance, many solvers have been developed for TV for algorithmic efficiency, including the primal-dual method [3], split-Bregmann method [9], and the alternating direction method of multipliers (ADMM). These solvers nevertheless cannot be easily extended for other efficacious priors such as mean curvature or Gaussian curvature [33], [14], [26], [11]. As a result, physically-natural priors representative for many imaging models and with advantageous properties and efficient solution strategies are an ever-existing need.

Among image priors, mean curvature (MC) and its variants are especially interesting, as mean curvature is the gradient of TV term and is further related with the classical mean curvature flow [4] and the Euler elastic bending energy [26]. A special case of Euler elasticity, called Willmore energy, is used in computer graphics for triangular mesh processing [6], [20]. Willmore energy is also preferred by most cell membranes [34], [27], demonstrating its biomechanical relevance.

In mathematics, MC is the average of all principal curvatures. It is also the average of all curvatures of cross-sectional curves created on that surface when it is cut at a point by all possible planes rotated around the surface normal at that point. This relates to the diffusion of heat-type equations, and in fluid mechanics, to the equilibrium of spherical droplets.

In this paper, we present a *weighted mean curvature* (WMC) prior that has several advantages properties, including sampling- and scale-invariance, its sparsity on natural images and relation to the gradient of a typical regularization. These properties are presented on example images and WMC is compared with traditional priors in numerical experiments on synthetic and real images. Since WMC does not assume the image smoothness, it is shown to successfully preserve sharp details in the resulting images. We further present a novel computation scheme to efficiently calculate WMC, including a neural-network implementation for GPU computations.

II. BACKGROUND AND MOTIVATION

Many image processing models can be expressed with the following variational framework:

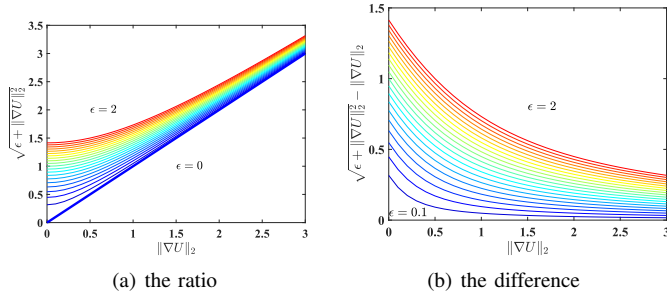$$U = \arg\min_U \left\{ \mathcal{E}(U(\vec{x}), f(\vec{x})) + \lambda \mathcal{R}(U(\vec{x})) \right\}, \quad (1)$$

Yuanhao Gong and Orcun Goksel are with Computer-assisted Applications in Medicine, Computer Vision Lab, ETH Zurich, Switzerland

Fig. 1. The role of ϵ in TV approximation, as ϵ is increased from 0 to 2 in increments of 0.1 (from blue to red curves).

where $\mathcal{E}$ is the imaging model, U is the unknown image to be estimated, f is the observed data, $\vec{x} \in R^n$ is the spatial coordinate, n is the image dimension, $\lambda > 0$ is a parameter (usually related with noise level), and $\mathcal{R}$ is the regularization term that imposes the assumed prior information. The prior term $\mathcal{R}$ is often independent from the imaging model $\mathcal{E}$; i.e., the same prior $\mathcal{R}$ can be used for various imaging models.

A. Total Variation Regularization

One of the most popular prior terms is TV regularization

$$\mathcal{R}_{\text{TV}} = \|\nabla U\|_2, \tag{2}$$

where ∇ is the gradient operator, and for most further notations, we drop $\vec{x}$ for simplicity. This regularization has many variants. One of them adopts ℓ_1 norm to impose anisotropy

$$\mathcal{R}_{\text{TVL1}} = \|\nabla U\|_1, \tag{3}$$

which prefers horizontal and vertical edges in U. To improve numerical stability, the following variant is also used:

$$\mathcal{R}_{\sim \text{TV}} = \int \sqrt{\epsilon + \|\nabla U(\vec{x})\|_2^2} \, d\vec{x}, \tag{4}$$

where $\epsilon > 0$ is a small real number. Although $\mathcal{R}_{\sim \text{TV}}$ numerically approximates $\mathcal{R}_{\text{TV}}$, these two terms are fundamentally different. This is demonstrated in Fig. 1 as ϵ is varied between 0 and 2 in increments of 0.1. Although, for small ϵ, this approximation error may be small at any given location $\vec{x}$, its integration on the whole imaging domain may add up to a substantial value, especially if the image area is large. This highlights the effect of ϵ beyond its numerical reasons.

For $\epsilon = 1$, the above becomes the area regularization

$$\mathcal{R}_{\text{area}} = \int \sqrt{1 + \|\nabla U(\vec{x})\|_2^2} \, d\vec{x}, \tag{5}$$

which imposes that U has a minimum area. Thus, this requires not only the gradient to be smooth, but also the image normal to be smooth. As a result, common staircase artifacts from TV can be significantly reduced using $\mathcal{R}_{\text{area}}$ [16].

Minimizing TV usually requires the gradient of $\mathcal{R}$ to be computed with respect to U [25], i.e.

$$-\frac{\partial \mathcal{R}_{TV}}{\partial U} = \frac{1}{2} \nabla \cdot \frac{\nabla U}{\|\nabla U\|_2}, \tag{6}$$

where $\nabla \cdot$ is the divergence operator. The right hand side of Eq. (6) is indeed the mean curvature of the isocontours of U.

Note that this computation as such would lead to numerical issues as $\|\nabla U\|$ vanishes. Note that the above definition of mean curvature for contours is different from that for surfaces or graphs, the latter of which was studied in earlier works [14], [12]. In this paper, we focus on the mean curvature definition commonly used in image community as in Eq. 6, and extend this to address its shortcomings with our proposed definition of *weighted mean curvature*.

B. Mean Curvature in Image Processing

Mean curvature (MC) has been used extensively in image processing problems [29], [18], [5], [14], [11], such as denoising [31], [33], registration [30], reconstruction [24], and decomposition [7]. For an image $U(\vec{x})$, the mean curvature of U at $\vec{x}$ is

$$H(U) = \frac{1}{n} \nabla \cdot \frac{\nabla U}{\|\nabla U\|_2}, \tag{7}$$

where ∇ and $\nabla \cdot$ are gradient and divergence operators, respectively. For $n = 2$, MC is equivalent to the right-hand side of Eq. (6) and we have

$$\begin{aligned} H &= \frac{U_x^2 U_{yy} - 2U_x U_y U_{xy} + U_y^2 U_{xx}}{2(U_x^2 + U_y^2)^{\frac{3}{2}}} \\ &= \frac{U_{yy} + U_{xx}}{2(U_x^2 + U_y^2)^{\frac{1}{2}}} - \frac{U_y^2 U_{yy} + 2U_x U_y U_{xy} + U_x^2 U_{xx}}{2(U_x^2 + U_y^2)^{\frac{3}{2}}}. \end{aligned} \tag{8}$$

This equation links MC with the Laplace operator and the diffusion along normal direction.

Mean curvature is independent of image contrast because $H(\alpha U) = H(U)$ for any scalar $\alpha \neq 0$. Thanks to this property, MC can provide a uniform regularization for images that contain objects of different contrast and thus is advantageous as image prior for regularization.

Mean curvature regularization is defined as

$$\mathcal{R}_H(U) = \int |H(U)|^q \, d\vec{x}, \tag{9}$$

where $q > 0$ is a scalar parameter; usually set to 1 or 2. If $\mathcal{R}_H(U) = 0$, the corresponding U is then a piecewise minimal surface (i.e., $\forall \vec{x}, H(U) = 0$).

Compared to TV regularization, $\mathcal{R}_H$ leads to better results for image denoising in practice [31], [32], which has been explained theoretically from a geometry point of view in [14], [11], [15] and a function analysis point of view in [12], [13].

C. Challenges with Mean Curvature

Despite its attractive features, application of MC has several difficulties in practice. First, the MC depends on scale and its use as regularization, i.e. $\mathcal{R}_H$, depends on sampling rate (described further in Section III-A). Thus, MC is not only affected by the geometry itself, but also by the sampling method and scale space, posing challenges as undesired side-effects in practical applications.

Second, minimizing the mean curvature regularization $\mathcal{R}_H$ is relatively challenging as it leads to a fourth-order partial differential equation [32]. Although several methods have been proposed, such as the multi grid method [2], augmented

Lagrange method [32], [24], and the fixed point method [28], to substantially reduced computations, their application on larger images is still far from practical, given realistic amount of computational resources.

Third, the discretization of Eq. (8) requires the first and second order derivatives to be approximated, for instance, by finite differencing, which is susceptible to noise and can be highly unstable. Further numerical issues arise for vanishing $\|\nabla U\|$ that appears in the denominator.

Finally, U needs to be assumed as smooth when numerically calculating its second order derivative. Although the mean curvature filter was proposed recently to relax this constraint via implicit minimization without computing the high-order derivatives [14], [12], the above computational challenges still persist.

D. Our Motivation and Contribution

To overcome the above challenges, we propose herein *weighted mean curvature* (WMC) to be used instead of mean curvature. WMC is fundamentally different from MC, with its following advantageous properties detailed further in the next section:

- **sampling-** and **scale-invariance,** crucial for images that contain objects with different scales and/or of different sampling rates.
- **sparsity on natural images:** As is later shown statistically on natural images, WMC is sparser than gradient, and thus would be preferred as a regularizer.
- **gradient of area regularization** can be numerically approximated using WMC, significantly simplifying the corresponding optimization procedure.
- **fast computation scheme:** For applying our weighted mean curvature on discrete images, we further propose herein a fast discrete computation scheme that can approximate WMC numerically. This has further advantages, including: (i) it does not require the signal to be second-order differentiable, hence better preserving edges; (ii) it is a very fast discrete operation, making WMC practical for most (even computatinally-demanding) image processing tasks; and (iii) it is numerical stable when $\|\nabla U\|_2$ vanishes and thus avoids any numerical issues, e.g. in contrast to those encountered with TV in Eq.(6) and MC in Eq.(7).

III. WEIGHTED MEAN CURVATURE

We define WMC as

$$H^w(U) = n\|\nabla U\|_2 H(U) = \|\nabla U\|_2 \left(\nabla \cdot \frac{\nabla U}{\|\nabla U\|_2} \right). \quad (10)$$

For 2D images $n = 2$, this equation becomes

$$H^w(U) = \underbrace{\Delta U}_{\text{isotropic diffusion}} - \underbrace{\frac{U_y^2 U_{yy} + 2U_x U_y U_{xy} + U_x^2 U_{xx}}{U_x^2 + U_y^2}}_{\text{diffusion along normal direction}}, \quad (11)$$

where Δ is the isotropic Laplace operator. Although this term still has the problem when $\|\nabla U\|_2 = 0$, its discrete
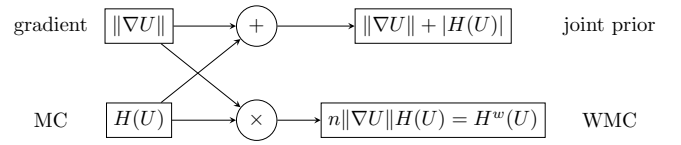

Fig. 2. Illustration of relationship between ∇U, $H(U)$, and $H^w(U)$.

computation can avoid such issue as shown later in Section IV. This WMC definition can be interpreted either as **gradient *weighted by* mean curvature** (weighted Total-Variation) or as **mean curvature *weighted by* gradient**. The relationship of WMC to gradient and MC is illustrated graphically in Fig. 2, in order to emphasize that the multiplication of the terms makes WMC fundamentally different from both gradient and mean curvature.

For WMC it is apparent that when *either* contributing term vanishes, their multiplication WMC will also go to zero. The small value of TV is a sufficient condition for the small value of WMC. Therefore, WMC is sparser i.e. is statistically more likely to be zero compared to the TV. We present a statistical comparison of these given natural images in Section III-D.

The above implies that WMC regularization is a superset of both $\mathcal{R}_H$ and $\mathcal{R}_{TV}$. WMC regularization term can be defined as

$$\mathcal{R}_{H^w}(U) = \int \left| H^w(U) \right|^q \mathrm{d}\vec{x} = \int \left| H(U) \|\nabla U\|_2 \right|^q \mathrm{d}\vec{x}, \quad (12)$$

where $q>0$ is a scalar parameter defining the norm. We use $q=1$ unless explicitly stated. Minimizing our WMC regularization requires either gradient or mean-curvature to be small. In other words, it will allow one of them to be large while the other one is close to zero. This behavior allows $\mathcal{R}_{H^w}$ to *automatically* choose gradient or MC or both to be minimized, leading to less artifacts compared to MC regularization $\mathcal{R}_H$ and TV regularization $\mathcal{R}_{TV}$.

In this paper, we introduce and study WMC as it can be used as both a regularization as well as the gradient of some regularization as shown in the following section. To that end, efficient numerical computation of WMC is essential, which is later addressed in Section IV. We also show below the sampling and scale invariance properties of WMC regularization. The gradient of WMC regularization, however, is not straight-forward to calculate, which is left for future work for potential application scenarios that may necessitate that. Nevertheless, we show several other practical applications in our results where WMC itself can be successfully computed and used.

A. Sampling- and Scale-Invariance

Note that while the MC regularizer in Eq.(9) is an integration of individual curvatures, the WMC regularizer we proposed is normalized (weighted) by divergence magnitude, cf. Eq.(12). Therefore, the spatial sampling effect is taken into account inherently by WMC. Similarly, the effect of scale is also incorporated through this weighting in WMC, which makes it a physically meaningful quantity. We further demonstrate this on a toy example below.
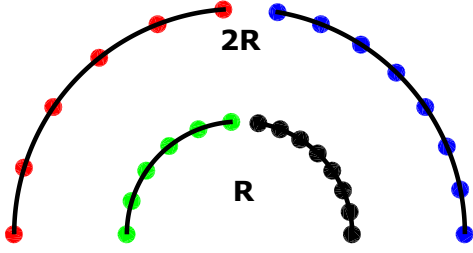
Fig. 3. Different sampling rates on half circles with radii R and $2R$.

Consider arcs with different sampling rates at two scales in Fig. 3, where gradients and curvatures of the arcs are computed at the given samples along these segments. For the inner (green and black) arcs of R radii, curvature H at the samples on the left and right segments are both equal, i.e. $\frac{1}{R}$. This then yields their integration $\mathcal{R}_H$ to be different as the number of integration points (samples) differ, i.e. $\frac{6}{R}$ vs. $\frac{8}{R}$ for the given examples, despite the fact that they present the same underlying geometry and continuum. In contrast, $\mathcal{R}_{H^w}$ yields the same value

$$\mathcal{R}_{H^w} \approx \sum_{i=1}^{6} \frac{1}{R} \frac{\pi R}{4 \times 6} = \sum_{i=1}^{8} \frac{1}{R} \frac{\pi R}{4 \times 8} . \tag{13}$$

demonstrating the sampling-invariance of WMC regularization $\mathcal{R}_{H^w}$.

Consider the scaled versions of these arcs, i.e. the outer segments with $2R$ radii. H would be reduced in both cases to $\frac{1}{2R}$ and thus each $\mathcal{R}_H$ also been halved. In contrast, H^w does not change as $\|\nabla U\|_2$ increases proportionally, demonstrating the scale-invariance of both H^w and $\mathcal{R}_{H^w}$.

The above presented shortcomings of mean-curvature is conventionally treated via heuristic-methods in image processing community, which is handled intrinsically by our WMC.

B. Contrast Invariance

Below we first analyze the classical TV model, showing MC as not contrast invariant. Consider a generic imaging model $AU = f$ where f is observed data, A is the imaging matrix, and U the image. This commonly has the solution strategy

$$U^\star = \arg\min_U \left\{ \frac{1}{2} \|AU - f\|_2^2 + \lambda \|\nabla U\|_2 \right\} . \tag{14}$$

The optimal solution $U^\star$ must satisfy Euler-Lagrange equation

$$A^T(AU^\star - f) - \lambda \nabla \cdot \frac{\nabla U^\star}{\|\nabla U^\star\|_2} = 0 \tag{15}$$

$$A^T(AU^\star - f) - n\lambda H(U^\star) = 0 . \tag{16}$$

Note that, since $H(\alpha U) = H(U)$, a scaled solution $\alpha U^\star$ does not satisfy the above for scaled data αf, for any $\alpha \neq 0$. Therefore, the above is not contrast invariant, i.e.

$$A^T(A\alpha U^\star - \alpha f) - n\lambda H(\alpha U^\star) \neq 0 . \tag{17}$$

Nevertheless, if MC in the above equation is replaced with WMC, contrast invariance can be shown as

$$A^T(A\alpha U^\star - \alpha f) - \lambda H^w(\alpha U^\star) = 0 . \tag{18}$$

Note that this is the Euler-Lagrange equation of *some* optimization form, i.e.

$$U^\star = \arg\min_U \left\{ \frac{1}{2} \|AU - f\|_2^2 + \lambda \mathcal{R}(U) \right\} . \tag{19}$$

given *some* regularization term $\mathcal{R}$. The form of such unknown regularization could be nontrivial; nevertheless for any solution strategy, its gradient is the main concern – which is defined above as WMC. In the following subsection, this unknown regularization term is shown to be a variant of area regularization and therefore WMC to be an approximation to the gradient of area regularization.

C. Approximate the Gradient of Area Regularization

Noticing the similarity between H^w in Eq. 11 and the gradient of area regularization in Eq. 5, i.e.

$$-\frac{\partial \mathcal{R}_{\text{area}}}{\partial U} = \sqrt{1 + |\nabla U|^2} \nabla \cdot \frac{\nabla U}{1 + |\nabla U|^2} \tag{20}$$

$$= \Delta U - \frac{U_y^2 U_{yy} + 2U_x U_y U_{xy} + U_x^2 U_{xx}}{1 + U_x^2 + U_y^2} \tag{21}$$

$$\approx \Delta U - \frac{U_y^2 U_{yy} + 2U_x U_y U_{xy} + U_x^2 U_{xx}}{U_x^2 + U_y^2} \tag{22}$$

$$= H^w , \tag{23}$$

we can consequently treat WMC as a numerical approximation to the gradient of the area regularization. This numerical approximation reduces computations significantly (thanks to the fast computation scheme presented in Section IV), which will enable many image processing tasks to apply area regularization in practical settings.

Although the approximation in (22) is less accurate when the gradient magnitude $|\nabla U|$ vanishes for constant regions of the image, in this case (20) becomes the Laplace operator, while H^w per definition in (10) also becomes the Laplace operator due to imposed Neumann boundary conditions; making these two quantities similar. We analyze WMC approximation to the gradient of the area regularization from two aspects: First, from the function analysis point of view, the approximation error is reduced when the gradient norm gets larger, as can be seen from the equations (21 and 22). Second, the relationship between $-\frac{\partial \mathcal{R}_{\text{area}}}{\partial U}$ and H^w can also be shown statistically, e.g. in Fig. 4 as computed on 500 natural images from BSDS500 dataset [1].

Based on the above approximation, the following model becomes *contrast-robust* (although not invariant due to the approximation)

$$U^\star = \arg\min_U \left\{ \frac{1}{2} \|AU - f\|_2^2 + \lambda \mathcal{R}_{\text{area}} \right\} . \tag{24}$$

Practical numerical applications of this model are shown in Section V-B.

D. Statistics of WMC on Natural Images

It is well-known that the gradient in natural images exhibits a heavy tail distribution. The mean curvature MC also presents a similar distribution [11]. Here, we show that WMC H^w
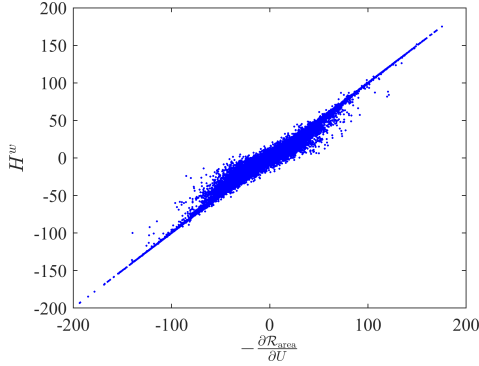
Fig. 4. The relationship between $-\frac{\partial \mathcal{R}_{\text{area}}}{\partial U}$ and H^w on 500 natural images from BSDS500 dataset.

satisfies a similar but even sparser distribution. We used 500 natural images from BSDS500 dataset. We computed (axis-aligned) gradients and WMC in these images, and in Fig. 5(a) we show the average gradient and average WMC distributions in log scale. It shows that WMC is sparser than gradient in this dataset. More specifically, let $p(H^w)$ and $p(\nabla U)$ denote the probability of WMC and the probability of gradient, respectively. Then, it can be seen that for any given real positive threshold v, the following holds: $p(|H^w| \leq v) > p(|\nabla U| \leq v)$. For example, for a threshold of $v = 30$, the probability of WMC is $p(|H^w| \leq 30) = 0.991$ while the probability of gradient is $p(|\nabla U| \leq 30) = 0.922$, as shown in Fig. 5(a). The sparsity can also be seen from the cumulative distributions in Fig. 5(b), where WMC raises tighter around zero, meaning it is more sparse (i.e. considering absolute values below any given threshold as practically zeros, then WMC will have more of those values). This fact also corroborates with the theoretical discussion earlier in the paragraph below Fig. 2 on observing zeros statistically.

Furthermore, the average WMC distribution can be modeled

(a) distributions in log scale

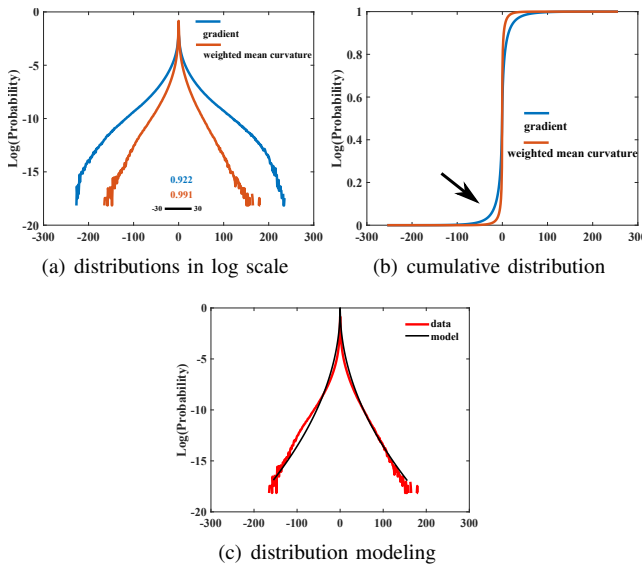(b) cumulative distribution

(c) distribution modeling

Fig. 5. Sparsity demonstration by (a) log-probability and (b) cumulative distributions of WMC and gradient, also (c) using a fitted model.

as

$$-\log(p(H^w)) \approx \frac{11}{8}|H^w|^{\frac{1}{2}}, \quad (25)$$

as shown the black line in Fig. 5(c), where p represents the probability. The power of $|H^w|$ term in this model fitting suggests that the parameter q in Eq. 12 should be close to $\frac{1}{2}$ when using $\mathcal{R}_{H^w}$ as regularization for natural images. Although $q = \frac{1}{2}$ is better from a modeling point of view, $q=1$ is preferred from optimization point of view, since $|\cdot|$ is easier than $\sqrt{\cdot}$ to minimize.

E. Connection with Mean Curvature Flow

Conventional mean curvature flow minimizes the total surface area by evolving a surface according to

$$\frac{\partial U(\vec{x}, t)}{\partial t} = H(U)\|\nabla U\|_2. \quad (26)$$

Note that the above is indeed a normalized form of WMC, i.e.

$$H(U)\|\nabla U\|_2 = \frac{1}{n}H^w(U). \quad (27)$$

With a discrete time step size of δ, we then have the iterations

$$U^{t+1} = U^t + \delta H^w(U^t). \quad (28)$$

This equation can be seen as minimizing the area regularization alone without a data fitting term. The bottleneck in performing this iteration is the evaluation of WMC, which conventionally requires an approximation of the first and second derivatives of U, which further suffers from numerical issues near $\|\nabla U\|_2 = 0$. Nevertheless, this is overcome by the novel computation scheme proposed herein to efficiently approximate WMC on images with a discrete Laplace operator.

IV. FAST DISCRETE COMPUTATION SCHEME FOR WMC

We first show the connection between WMC and Laplace operator, and next analyze discrete Laplace kernels and normal directions. We then present the combination of a regression kernel with discrete normal directions as a fast computation scheme.

A. Discretization of Weighted Mean Curvature

Given the WMC expression in Eq.(11), the first term can be approximated by a Laplace operator and the second term is the diffusion along the normal direction.

1) Discrete Isotropic Laplace Operator: Let us represent the Laplace operator with a convolution kernel, i.e. $\Delta U = k * U$. We compare the following four discrete convolution kernels as promising Laplace operator options for 2D images:

$$k_1 = \begin{bmatrix} \frac{1}{8} & \frac{1}{8} & \frac{1}{8} \\ \frac{1}{8} & -1 & \frac{1}{8} \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} \end{bmatrix}, \quad k_2 = \begin{bmatrix} -\frac{1}{16} & \frac{5}{16} & -\frac{1}{16} \\ \frac{5}{16} & -1 & \frac{5}{16} \\ -\frac{1}{16} & \frac{5}{16} & -\frac{1}{16} \end{bmatrix}, \quad (29)$$

$$k_3 = \begin{bmatrix} \frac{1}{20} & \frac{1}{5} & \frac{1}{20} \\ \frac{1}{5} & -1 & \frac{1}{5} \\ \frac{1}{20} & \frac{1}{5} & \frac{1}{20} \end{bmatrix}, k_4 = \begin{bmatrix} \frac{1}{12} & \frac{1}{6} & \frac{1}{12} \\ \frac{1}{6} & -1 & \frac{1}{6} \\ \frac{1}{12} & \frac{1}{6} & \frac{1}{12} \end{bmatrix}. \quad (30)$$

Kernel k_1 is a common isotropic Laplace operator. Kernel k_2 is from the mean curvature filter in [14]. Kernel k_3 originates
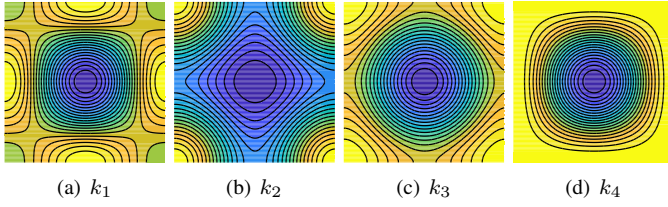
Fig. 6. Isotropy of Laplace kernels shown with Fourier spectral magnitude.

from the numerical analysis field [23]. Kernel k_4 is common in the image processing community [21]. To analyze the isotropy of these kernels, we transformed them into Fourier domain with their spectral magnitude plotted in Fig. 6 with isolines.

Kernel k_4 is seen to be the most isotropic one and is thus used in this work as the discrete Laplace operator.

2) Discrete Normal Direction: Since gradient, MC and WMC are local properties, as their support region gets smaller, they provide better approximation, especially at sharp edges (although this may differ in smoother regions). In this paper, we choose a 3×3 window and consider only 8 possible normal directions separated by $45°$ indicated with half windows shown in Fig. 7. At any location $\vec{x}$, its normal $\vec{n}$ is then approximated by one of these eight cases $\{\vec{n}_i | i = 1, .., 8\}$. Note that these directions include the horizontal and vertical gradients typically used in (anisotropic) TV regularization.
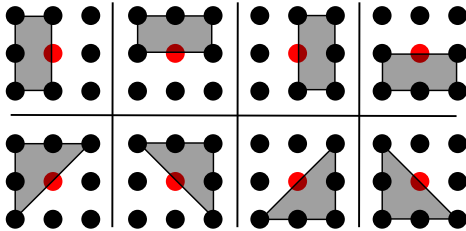

Fig. 7. Eight cases $\{\vec{n}_i | i = 1, .., 8\}$ of half window regression

The limitation of this discrete normal assumption is that the continuous normal $\vec{n}$ is only approximated by the discrete $\vec{n}_i$, with potential approximation errors. Nevertheless, with a limited discrete directions as in Fig. 7, the normal diffusion term of H^w in Eq.(11) becomes enumerable such that its computation is fast and straight-forward. Additionally, by replacing this normal diffusion term with regression of half windows, *any numerical instability due to $\|\nabla U\| = 0$ is avoided*. Moreover, since the diffusion can only happen along edges (but not normal to edges), this discretization also helps to preserve edges. This can be envisioned as an extreme case of anisotropic diffusion.

For the half-window regressions, we use Neumann boundary conditions. This imposes the reflection symmetry property in our convolution kernels [15]. This boundary condition is imposed at every pixel, which is different from traditional boundary conditions that are only enforced at large gradient locations.

B. Convolution Scheme

Combining the proposed discrete Laplace kernel with the 8 half-windows with Neumann boundary conditions [15] yields the following eight convolution kernel candidates:

$$h_1 = \begin{bmatrix} \frac{1}{6} & \frac{1}{6} & 0 \\ \frac{1}{3} & -1 & 0 \\ \frac{1}{6} & \frac{1}{6} & 0 \end{bmatrix}, \quad h_2 = \begin{bmatrix} \frac{1}{6} & \frac{1}{3} & \frac{1}{6} \\ \frac{1}{6} & -1 & \frac{1}{6} \\ 0 & 0 & 0 \end{bmatrix},$$

$$h_3 = \begin{bmatrix} 0 & \frac{1}{6} & \frac{1}{6} \\ 0 & -1 & \frac{1}{3} \\ 0 & \frac{1}{6} & \frac{1}{6} \end{bmatrix}, \quad h_4 = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{6} & -1 & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{3} & \frac{1}{6} \end{bmatrix},$$

$$h_5 = \begin{bmatrix} \frac{1}{6} & \frac{1}{3} & \frac{1}{12} \\ \frac{1}{3} & -1 & 0 \\ \frac{1}{12} & 0 & 0 \end{bmatrix}, h_6 = \begin{bmatrix} \frac{1}{12} & \frac{1}{3} & \frac{1}{6} \\ 0 & -1 & \frac{1}{3} \\ 0 & 0 & \frac{1}{12} \end{bmatrix},$$

$$h_7 = \begin{bmatrix} 0 & 0 & \frac{1}{12} \\ 0 & -1 & \frac{1}{3} \\ \frac{1}{12} & \frac{1}{3} & \frac{1}{6} \end{bmatrix}, h_8 = \begin{bmatrix} \frac{1}{12} & 0 & 0 \\ \frac{1}{3} & -1 & 0 \\ \frac{1}{6} & \frac{1}{3} & \frac{1}{12} \end{bmatrix}.$$

From these kernels, we compute eight signed distances

$$d_i = h_i * U, \quad \forall i = 1, .., 8 . \tag{31}$$

Resulting $\{d_i\}$ can be interpreted as the signed projection distances to the hyperplanes defined by each half-window. Therefore, the d_i that has the minimum absolute value is the proximal projection distance that has the highest probability to a minimal surface. We use it as our estimation for WMC, i.e.

$$H^w \approx d_m, \quad \text{where} \quad m = \arg\min_i \{|d_i|; i = 1, .., 8\} . \tag{32}$$

We call the algorithm described by Eq. (31) and (32) as Half Laplace Operator because the support region is a half window.

C. Neural Network Representation

This computation scheme has a convolutional neural network representation. The convolutions in Eq. 31 can be represented as convolution layers in a neural network structure while Eq. 32 acts as a nonlinear activation function. Thus, our discrete computation scheme can be interpreted as a neural network. For instance, an architecture for computing mean curvature is shown in Fig. 8. The step symbol in Fig. 8 indicates Eq. 32 and is the activation function in neural networks.
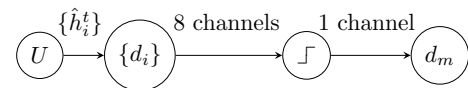

Fig. 8. Neural network representation of our scheme

Based on such neural-network representation, two further extensions of our discrete computation scheme can be envisioned. First, $\{h_i\}$ could be extended as *learnable* kernels $\{\hat{h}_i^t\}$ (t is the layer index) for a specific dataset, such as natural images or medical images of a certain modality and given anatomy; such that these learned kernels would be more effective than the general $\{h_i\}$. Since $\{\hat{h}_i^t\}$ are coupled with the kernels $\{\hat{h}_i^{t-1}\}$ and $\{\hat{h}_i^{t+1}\}$ in the previous and next
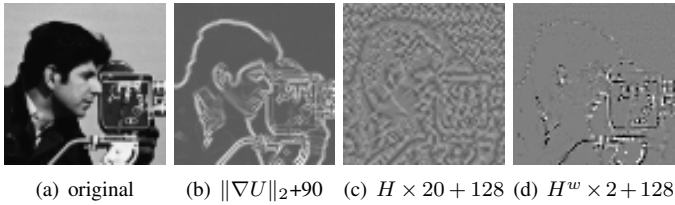
| (a) original | (b) $\|\nabla U\|_2 + 90$ | (c) $H \times 20 + 128$ | (d) $H^w \times 2 + 128$ |

Fig. 9. A close-up of (a) an image; (b) its gradient norm, (c) mean curvature, and (d) WMC; all scaled to [0,255] gray value range as shown.

layers, their layer specific setting might be more efficient in minimizing the loss function. Second, the number of kernels can be increased, allowing for more than 8 *normal* directions or possible operator structures. We present numerical results from a neural network representation of our computational scheme later in Section V-E.

D. Support Region and Computational Complexity

The presented computation scheme has linear computational complexity and is computationally very efficient, with an implementation presented in the Appendix. The eight convolution operations can also be implemented in parallel on modern hardwares, such as Graphic Processing Unit (GPU). Our parallel implementation can process 33.2 Giga-pixels per second on a TITAN X Pascal GPU, using its native CUDA language. This indicates that our proposed methods can be used for images with very large sizes or in real-time image processing tasks. Furthermore, since the half windows in Fig. 7 have overlapping regions, it is also possible to use integral images to further improve computational performance.

Since mean curvature is a local property, smaller support region is always preferred (as well resulting in a faster computation). Although the continuous definition of H can represent edges from a mathematical point of view, its finite difference approximation in Eq. 8 requires a relatively larger support region (5×5 or larger depending on the finite difference scheme used) and thus cannot preserve edges due to such larger support region. The support region for our proposed scheme has only 6 taps (half-windows seen in Fig. 7) and thus WMC H^w can preserve edges better than MC H. This is demonstrated in the next section.

V. EXPERIMENTS

A. Comparison between $\|\nabla U\|$, H and H^w

In order to show the difference between gradient $\|\nabla U\|$, mean-curvature H, and WMC H^w, we apply them on a sample image, where a patch is shown in Fig. 9 as a close-up. Since WMC has a smaller support region, it captures local geometries better than MC; e.g. the variance of H^w is small in the sky, while H still presents strong local variance in Fig. 9.

B. Comparison of $\mathcal{R}_{TV}$, $\mathcal{R}_H$, and $\mathcal{R}_{area}$

We compare three different regularization options for the same imaging problem

$$U^\star = \arg\min_U \left\{ \frac{1}{2}\|AU - f\|_2^2 + \lambda\mathcal{R}(U) \right\}, \quad (33)$$

where $\mathcal{R}$ is one of $\mathcal{R}_{TV}$, $\mathcal{R}_H$ and $\mathcal{R}_{area}$. For simplicity, we set A as identity matrix and checked the behaviors of these regularizations at different λ. We used a numerical ground-truth image seen in Fig. 10(a), where each row of circles indicate scale change and each column indicate contrast difference.

For $\mathcal{R}_{TV}$ case, we solve it using [19] with an arbitrarily high 50 K iterations. We set $\lambda = 300$, 700, and 1000 with the resulting $U^\star$ seen in Fig. 10(b-d). Herein relatively large values of λ were selected to demonstrate the effect of and artifacts from regularization at its extreme. When λ is increased, staircase artifacts are observed and the contrast is lost, where the smaller circles with lower contrast get smoothed out first. Note that although the TV model can capture sharp edges ideally [22], it may lose contrast, especially at high regularization weights, as seen in cross-sections in Fig. 10(e and i).

For $\mathcal{R}_H$ case, we solve it using [2]. We set $\lambda = 20$, 150, and 1000, with the results seen in Fig. 10(f-h). When λ is increased, resulting images get smoother. Since mean-curvature H is contrast invariant, the results are robust to contrast change, but not robust to scale change.

For area regularization, we use gradient descent for the data fitting term with our approximation Eq. 20 for $\mathcal{R}_{area}$. More specifically, we apply the following iteration

$$\frac{\partial U}{\partial t} = -A^T(AU - f) - \lambda\frac{\partial\mathcal{R}}{\partial U}, \quad (34)$$

where $-\frac{\partial\mathcal{R}_{area}}{\partial U} \approx H^w(U)$. We set $\lambda = 20$, 150, and 200, with the results seen in Fig. 10(j-l). Note that with increasing λ, the shrinking and smoothing behavior is the same across all scale and contrast levels. This empirical observation corroborates the contrast invariance discussion of area regularization earlier in Section III-C. Since the normal directions are limited to 8 possible cases, WMC results at high regularization weights might get axis-biased; nonetheless this artifact is not distracting nor often observable in our results, as demonstrated in the next sections.

Fig. 10(e) shows the contrast profiles for the three methods above along the first column of circles, and similarly Fig. 10(i) along the first row of circles. Our WMC results are seen to be the closest to the ground-truth.

C. Data Fidelity with ℓ_1 Norm

We herein study the following ℓ_1-based variation model, which is known to treat Laplace noise well and is robust to outliers while imposing sparsity:

$$\min_U \left\{ \|AU - f\|_1 + \lambda\mathcal{R}(U) \right\}. \quad (35)$$

We show the solution of this model based on our WMC-based area regularization, using the following primal-dual method.

For $b = AU - f$, we then have the following model

$$\min_U \left\{ \|b\|_1 + \lambda\mathcal{R}(U) \right\} \ s.t. \ b = AU - f. \quad (36)$$

This is equivalent to the dual representation

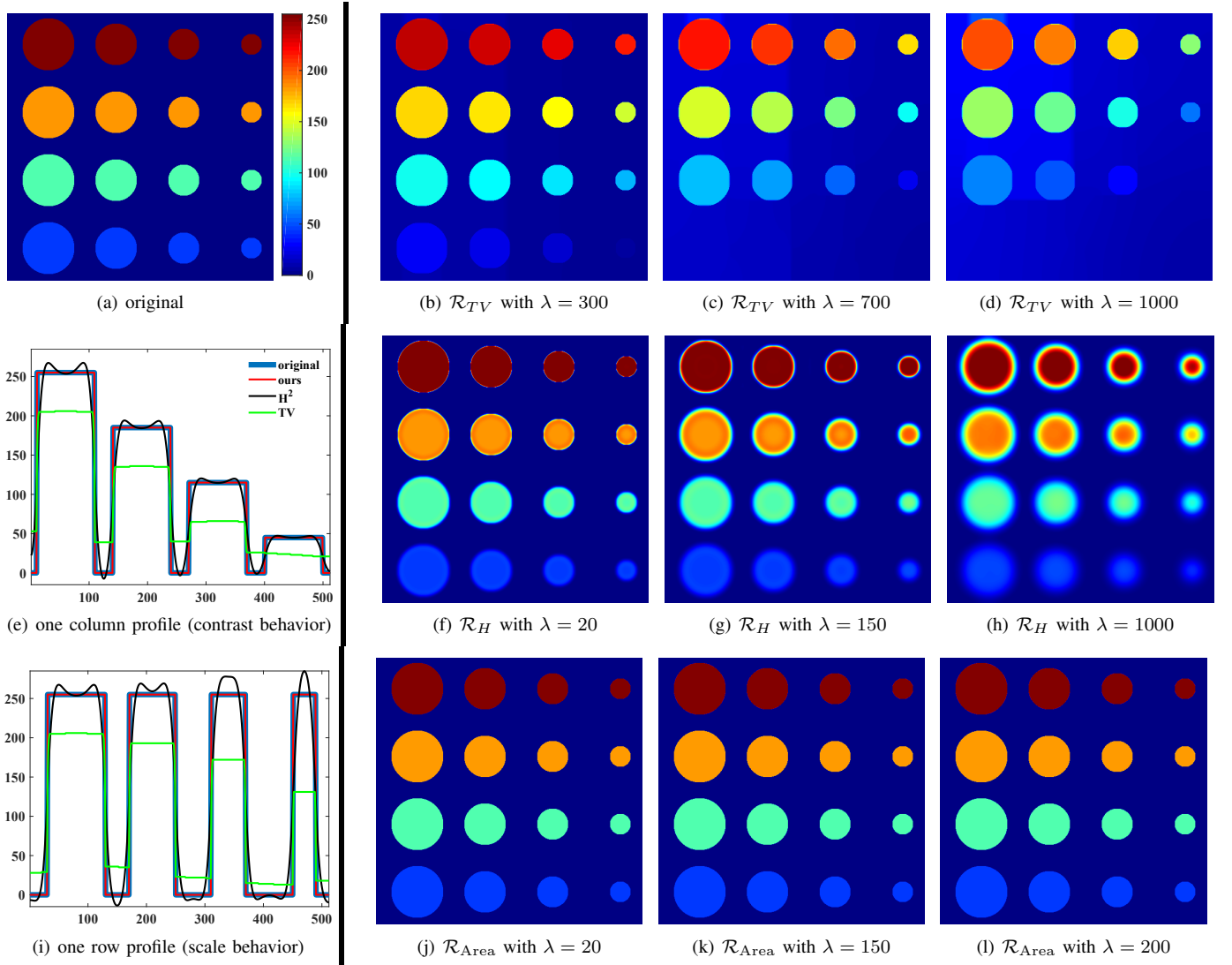$$\min_U \left\{ \|b\|_1 + \lambda\mathcal{R}(U) + \frac{1}{2\alpha}\|b - AU + f - d\|_2^2 \right\}, \quad (37)$$

Fig. 10. Comparison of $\mathcal{R}_{TV}$, $\mathcal{R}_H$ and $\mathcal{R}_{\text{Area}}$. (a) original image; (b), (c), and (d) are results from TV regularization; (f), (g), and (h) are results from mean curvature regularization; (j), (k), and (l) are results from area regularization. (e) and (l) show the cross-sectional profiles for the first column and first row inclusions, respectively, in (d), (h), and (l).

where $\alpha > 0$ is a small constant and d is a scaled dual variable.

Within an iterative solution, at each iteration, given $r \equiv AU^t - f + d^t$ there is a closed-form solution for b as

$$
b = \begin{cases} r - \alpha, & r > \alpha \\ 0, & |r| \le \alpha \\ r + \alpha, & r < -\alpha \end{cases}, \tag{38}
$$

and for d as

$$
d^{t+1} = d^t + AU^t - f - b = \begin{cases} \alpha, & r > \alpha \\ r, & |r| \le \alpha \\ -\alpha, & r < -\alpha \end{cases}. \tag{39}
$$

For U, the following problem needs to be solved

$$
\min_U \left\{ \lambda \mathcal{R}(U) + \frac{1}{2\alpha} \| b - AU + f - d^{t+1} \|_2^2 \right\}. \tag{40}
$$

for which one way is to use gradient descent as follows:

$$
\begin{aligned} \frac{\partial U}{\partial t} &= -\lambda \frac{\partial \mathcal{R}}{\partial U} - \frac{1}{\alpha} A^T (AU^t - f + d^{t+1} - b) \\ &= -\lambda \frac{\partial \mathcal{R}}{\partial U} - \frac{1}{\alpha} A^T (2d^{t+1} - d^t). \end{aligned} \tag{41}
$$

As a result, b is eliminated from the optimization procedure, d and U need to be updated alternately until convergence. As only a matrix multiplication is needed each for d^{t+1} and U^{t+1} updates, the iterations can be computed very fast.

For area regularization as in Eq. 20, we then have

$$
\frac{\partial U}{\partial t} \approx \lambda H^w(U^t) - \frac{1}{\alpha} A^T (2d^{t+1} - d^t). \tag{42}
$$

We can solve this equation using our computation scheme for H^w. When A is the identity matrix, this model leads to a smoothing problem. Results of this model for different λ are shown in Fig. 11 with a qualitative assessment on the preservation of most major structures even at higher λ.

(a) original input f (b) $\lambda=1$, SSIM=0.9106   (c) $\lambda=3$, SSIM=0.8610   (d) $\lambda=10$, SSIM=0.8171   (e) $\lambda=20$, SSIM=0.8000

(f) $f-U+128$ when $\lambda=1$ (g) $f-U+128$ when $\lambda=3$ (h) $f-U+128$ when $\lambda=10$ (i) $f-U+128$ when $\lambda=20$
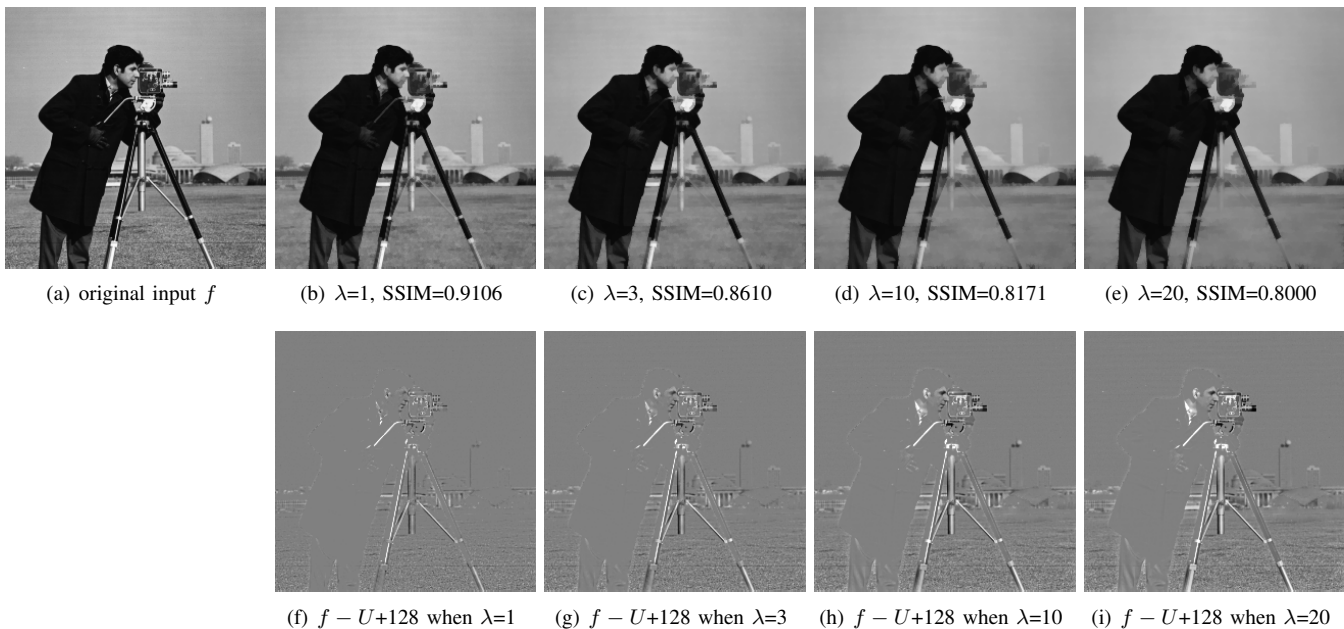
Fig. 11. The effect of increasing λ, where the second row shows the residual $f-U$. Qualitatively speaking, with $\lambda=1$ the handle of the camera is still visible, while with $\lambda=3$ the handle disappears. With $\lambda=10$ the roof of the tall building is no more visible. With $\lambda=20$ the results do not further change significantly, while major structures are still preserved to a large extent.

We used structural similarity index measurement (SSIM) to quantify the structural similarity between images.

D. Mean Curvature Flow with Edge Preservation

Given the advantages of WMC, we herein show that its proposed computation scheme is ideal, yielding significantly further advantages in its applications. We study the iterative mean curvature flow application in Eq.(28), which is independent of an imaging modal and data term. We discretize WMC using the conventional method or our proposed computation scheme, and present results at different iterations t in Fig. 12. As seen in Fig. 12, our discrete computation scheme can preserve local geometry and sharp edges much better compared to the traditional discretization. This is due to two reasons: (i) our method is performed within a smaller support with limited diffusion directions; and (ii) our computation does not require the image to be second-order differentiable, whereas the traditional discretization computes second derivatives hence requiring the result to be at least second-order smooth within given supports.

A close-up is shown in Fig. 12(e) to demonstrate the smoothing-while-edge-preserving behavior of our computation scheme. With increasing iteration number, the wrinkles on the flag are correctly smoothed out, while the edges of cross are well preserved. A similar optimal behavior is also observed on the mountain and snow textures. In contrast, with traditional mean-curvature computation, the sharp corners become rounded while most structural information is also smoothed as Fig. 12(d).

Note that the above is merely a difference from the two approximations (computation schemes) of the same WMC operation. We conclude that our proposed discrete computation is potentially more suitable for most image processing tasks.

E. Convolutional Neural Network Implementation

As mentioned, our computation scheme has a neural network representation, which can be used to find data adaptive kernels $\{\hat{h}_i\}$. Using our neural network structure in Fig. 13, we solve the following model

$$\min_{U} \left\{ \frac{1}{s}\|AU - f\|_s^s + \lambda \mathcal{R}_{\text{area}}(U) \right\}, \tag{43}$$

for $s=1$ and $s=2$. For simplicity, we again assume A as identity. Traditional gradient decent method for this model required hundreds of iterations to converge, whereas our neural network method was able to obtain acceptable results with only two layers and eight learned filters. Neural network implementation thus required two to three orders of magnitude shorter time than the iteration method, when both are performed on the same hardware.

To obtain the learned kernels, we extracted 10,000 image patches of 128×128 pixels each, randomly from natural image dataset BSDS500. We used the network structure shown in Fig. 13, with two such consecutive layers (more layers did not reduce the loss function anymore for this problem). We separately learned 4, 8, and 16 convolutional filters of 3×3 size each. We set a learning rate of 2×10^{-4}, batch size of 32, and regularization λ of 5. The eight learned kernels in the first layer from our 8 filter network are shown in Fig. 14. Some of these kernels already look like our half Laplace kernels. The average loss function for different number of learned filters is tabulated in Table I, in comparison to the iterative method with 1000 iterations. Note that merely a two-layer network yields better results (energy-level and quality) compared to the iterative approach, since these filters are now data-adaptive. Since the filters $\{\hat{h}_i^t\}$ are independent from image resolution, they can be trained on low resolution images. Note that the

(a) original   (b) deriv, t=10: SSIM=0.923   (c) deriv, t=50: SSIM=0.870   (d) deriv, t=200: SSIM=0.821

(e) patch details   (f) HalfLaplace, t=10: SSIM=0.974   (g) HalfLaplace, t=10^2: SSIM=0.940   (h) HalfLaplace, t=10^3: SSIM=0.909
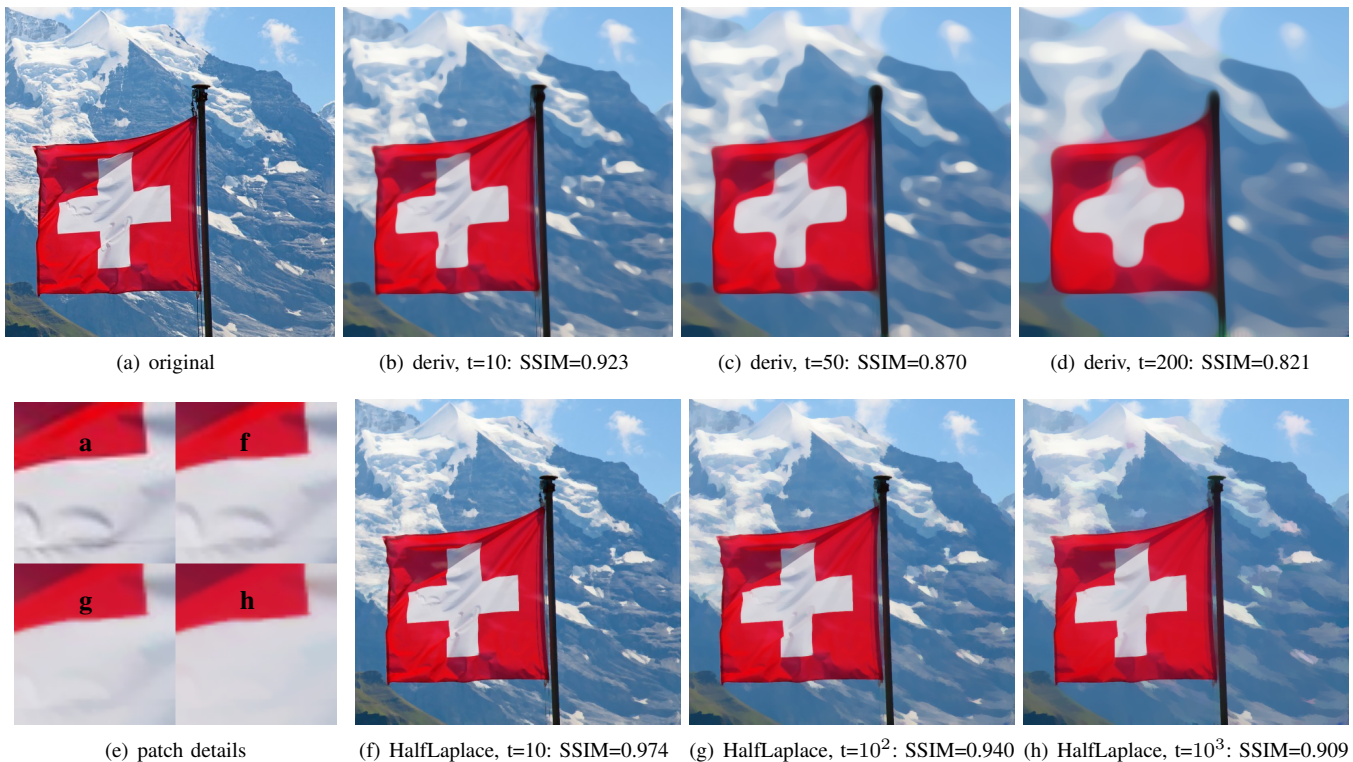
Fig. 12. Comparison of Mean Curvature Flow with the standard computation scheme (first row) and the proposed computation scheme (second row). Detailed patches from our method are shown in (e), where original is at the left up, t=10 at the right up, t=100 at left bottom and t=1000 at the right bottom. Our scheme preserves the edges and sharp corners (cf. the flag and the cross). Mean curvature flow was performed on each color channel separately.
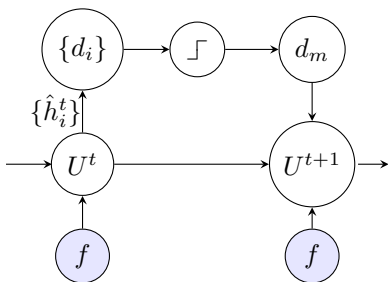

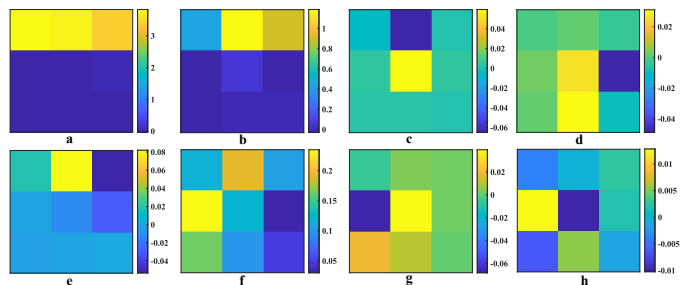Fig. 13. Neural network implementation of variational model in Eq. (43)

Fig. 14. Learned kernels $\{\hat{h}_i^t\}$ in the first layer. Some of these kernels already look like half windows (e.g., a, b, e, g, h) and gradient operators (c, d).

TABLE I
AVERAGE ENERGY LEVEL FOR DIFFERENT SOLVERS.

Our Neural Network Method			Iterative Method
# filters = 4	8	16	iteration=1000
$s=1$	35.0	33.3	32.7	39.4
$s=2$	66.5	66.3	66.1	69.8

content of testing images should be similar with the training dataset, i.e. natural scenery in this case. Nevertheless, while these learned filters would be valid only for training-like input, the generic kernels given in Section IV-B are valid for any input data.

Figure 15 shows some example results from our neural network implementation with $s = 2$, $\lambda = 5$, and 8 kernels used. The details are seen to be removed, while the main structures being successfully preserved. With TensorFlow li-

brary and Python implementation, each prediction took 1.2 ms on a GeForce 940MX (384 CUDA cores) with Windows 10 (Thinkpad T480) and 0.25 ms on a GeForce GTX TITAN X (3072 CUDA cores) with Linux. Better performance can be obtained by using native CUDA C++ language. A two-layer network is seen to be orders of magnitude faster than the iterative structure.

F. Compared with Other Edge Preserving Filters

One iteration of the mean curvature flow with our discrete computation scheme can be considered as a filter. Therefore, our filter is comparable with other edge preserving filters, such as Domain Transform (DT) [8] and Guided Filter (GF) [17]. We tested these three filters on four images as shown in Fig. 16. As seen with these results, DT tends to generate block artifacts while GF may lose the color contrast. Our method
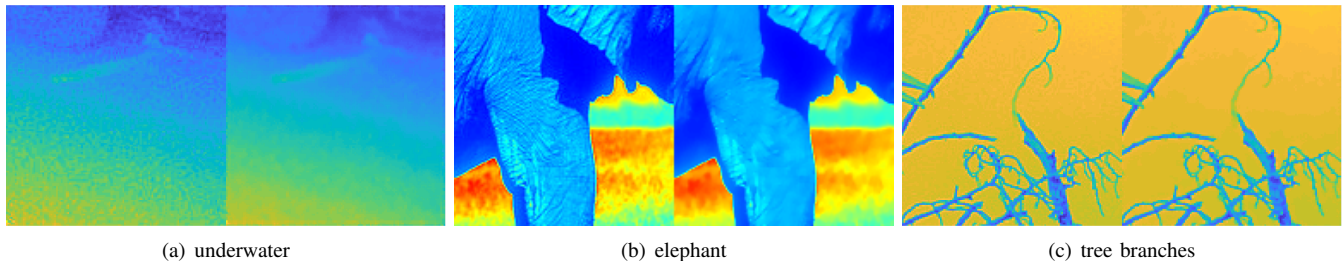
(a) underwater (b) elephant (c) tree branches

Fig. 15. In each sub-image, a close-up of (left) the original image and (right) the output from our neural network implementation with $s=2$, eight kernel filters, $\lambda=5$ are shown. (a) Image is smoothed while preserving edges. (b) Wrinkles on the skin are removed, while elephant outline is well preserved. (c) Contrast is well preserved while smoothing the background and the branches.
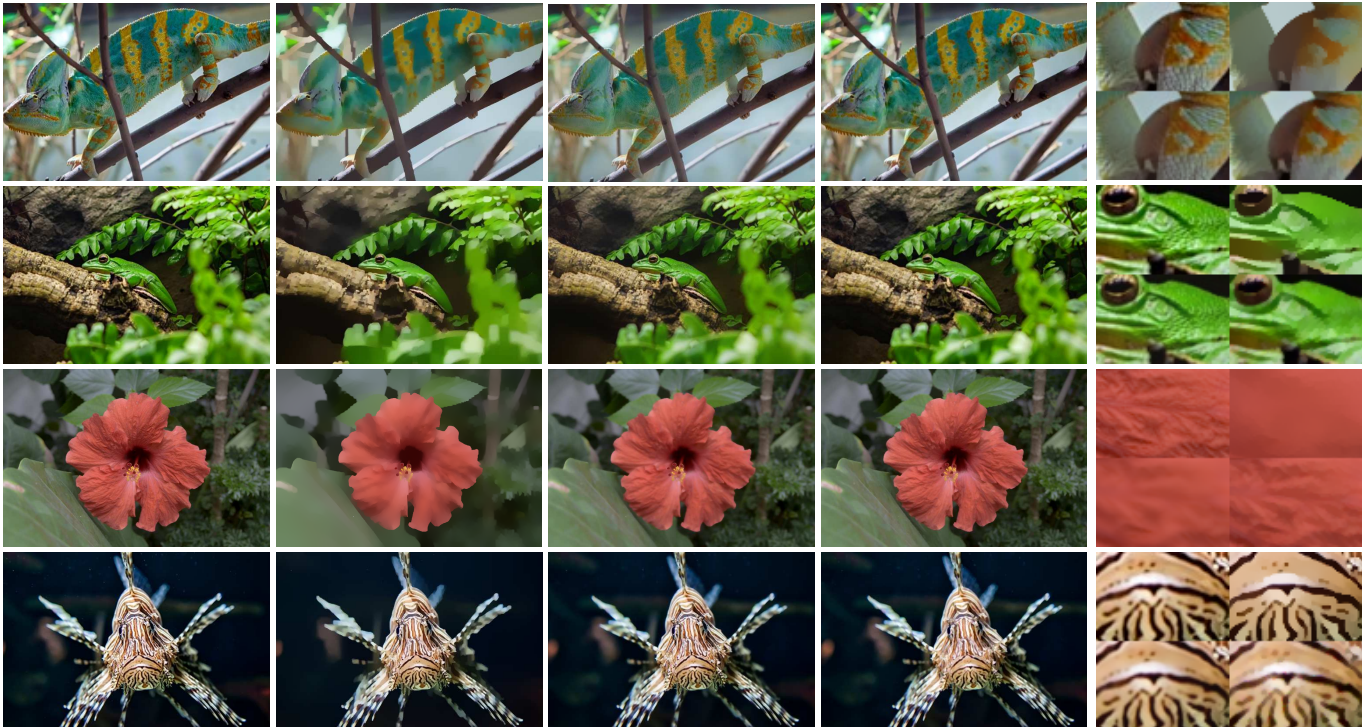

Fig. 16. Comparison with other edge preserving filters. From left to right: original, Domain Transform [8], Guided Filter [17], our method with 10 iterations, and close-ups of the results (in the same order, row-first from top-left to bottom-right). All method parameters were set as in their original papers. In image 1 at the foot region, DT smooths all details and generates block artifacts, GF loses the color contrast, while our proposed method successfully removes the texture detail preserving major structures. Similarly, our method successfully removes the details in image 2 without generating block artifacts nor losing color contrast; preserves the color contrast in image 3 the best; and removes the dots in image 4 while keeping the stripes.

successfully removes the details while preserving the major structures, without any such side-effects.

VI. CONCLUSION

We presented weighted mean curvature and showed its advantageous properties such as scale- and sampling-invariance. It can be used as an approximation of gradient for area regularization, which facilitates optimization in image processing applications. We discretized the computation with 8 kernels with directional normals, which avoids numerical issues and leads to a very efficient computation scheme. Our experiments confirm the benefits from the proposed WMC and its discrete computation scheme, including scale- and contrast-robustness, fast computation, and edge preservation. We compared mean curvature flow computed by a classical method and our new scheme, demonstrating better edge preservation with the latter.

We also presented our scheme as a neural network layer with its performance on natural images and potential to learn these filters. In comparison to other edge preserving filters, our method can produce higher quality results.

As being independent from any particular imaging model, WMC and area regularization can both be applied on a large range of image processing problems, including smoothing, denoising, super resolution, deconvolution, and image reconstruction. In most of these problems, minimizing the area regularization becomes the bottleneck, where our efficient WMC approximation can significantly accelerate this procedure. Our computation scheme can be further extended for higher dimensional data such as video and 3D images. Thanks to the high performance, our method can be adopted for real time image processing tasks or used on embedded devices such as FPGA, with potential applications in mobile applications

such as on smart phones, medical devices, and microscopes.

ACKNOWLEDGMENTS

APPENDIX

```
function Hw=WeightedMeanCurvature(u)
%compute weighted mean curvature
k1=[1,1,0;  2, -6,0;  1,1,0]/6;
k2=[2,4,1;  4,-12,0;  1,0,0]/12;
dist = zeros([size(u),8],'single');
dist(:,:,1) = conv2(u,k1,'same');
dist(:,:,2) = conv2(u,fliplr(k1),'same');
dist(:,:,3) = conv2(u,k1','same');
dist(:,:,4) = conv2(u,flipud(k1'),'same');
dist(:,:,5) = conv2(u,k2,'same');
dist(:,:,6) = conv2(u,fliplr(k2),'same');
dist(:,:,7) = conv2(u,flipud(k2),'same');
dist(:,:,8) = conv2(u,rot90(k2,2),'same');
%% find minimum signed distance
[~,ind] = min(abs(dist),[],3);
%turn sub to index (sub2ind but faster)
N=size(u,1)*size(u,2);
offset=int32(reshape(-(N-1):0,size(u,1),size(u,2)));
index=offset+int32(ind*N);
Hw = dist(index);
```

REFERENCES

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898–916, May 2011.

[2] C. Brito-Loeza and K. Chen, "Multigrid algorithm for high order denoising," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 363–389, 2010.

[3] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, pp. 120–145, 2011.

[4] Y. G. Chen, Y. Giga, and S. Goto, "Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations," *J. Differential Geom.*, vol. 33, no. 3, pp. 749–786, 1991.

[5] A. Ciomaga, P. Monasse, and J. M. Morel, "Level lines shortening yields an image curvature microscope," in *2010 IEEE International Conference on Image Processing*, Sept 2010, pp. 4129–4132.

[6] K. Crane, U. Pinkall, and P. Schröder, "Robust fairing via conformal curvature flow," *ACM Trans. Graph.*, vol. 32, 2013.

[7] S. D. El Hadji, R. Alexandre, and A.-O. Boudraa, "2d curvature-based analysis of intrinsic mode functions," *IEEE Signal Processing Letters*, vol. PP, no. 99, pp. 1–1, August 2017.

[8] E. S. L. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM TOG*, vol. 30, no. 4, pp. 69:1–69:12, 2011, proceedings of SIGGRAPH 2011.

[9] T. Goldstein, X. Bresson, and S. Osher, "Geometric applications of the split Bregman method: Segmentation and surface reconstruction," *J. Sci. Comput.*, vol. 45, no. 1, pp. 272–293, 2009.

[10] Y. Gong and I. Sbalzarini, "A natural-scene gradient distribution prior and its application in light-microscopy image processing," *IEEE J Selected Topics in Signal Processing*, vol. 10, no. 1, pp. 99–114, 2016.

[11] Y. Gong, "Spectrally regularized surfaces," Ph.D. dissertation, ETH Zurich, Nr. 22616, 2015, http://dx.doi.org/10.3929/ethz-a-010438292.

[12] ——, "Bernstein filter: A new solver for mean curvature regularized models," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 1701–1705.

[13] Y. Gong and I. F. Sbalzarini, "Local weighted Gaussian curvature for image processing," *Intl. Conf. Image Proc. (ICIP)*, pp. 534–538, 2013.

[14] ——, "Curvature filters efficiently reduce certain variational energies," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1786–1798, 2017.

[15] Y. Gong, Q. Wang, C. Yang, Y. Gao, and C. Li, "Symmetry detection for multi-object using local polar coordinate," *Lecture Notes in Computer Science*, vol. 5702, p. 277, 2009.

[16] G. Graber, J. Balzer, S. Soatto, and T. Pock, "Efficient minimal-surface regularization of perspective depth maps in variational stereo," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 511–520.

[17] K. He, J. Sun, and X. Tang, "Guided image filtering," *ECCV 2010*, pp. 1–14, 2010.

[18] G. Huisken and T. Ilmanen, "The inverse mean curvature flow and the riemannian penrose inequality," *J. Differential Geom.*, vol. 59, no. 3, pp. 353–437, 11 2001.

[19] R.-Q. Jia and H. Zhao, "A fast algorithm for the total variation model of image denoising," *Advances in Computational Mathematics*, vol. 33, no. 2, pp. 231–241, 2010.

[20] C. Jing-yi, "The willmore functional of surfaces," *Appl. Math.-J. Chin. Univ. Ser. B*, vol. 28, no. 4, pp. 485–493, December 2013.

[21] B. Kamgar-Parsi, B. Kamgar-Parsi, and A. Rosenfeld, "Optimally isotropic laplacian operator," *IEEE Transactions on Image Processing*, vol. 8, no. 10, pp. 1467–1472, Oct 1999.

[22] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 460–489, 2005.

[23] M. Patra and M. Karttunen, "Stencils with isotropic discretization error for differential operators," *Numerical Methods for Partial Differential Equations*, vol. 22, no. 4, pp. 936–953, 2006.

[24] F. Ren, T. Qiu, and H. Liu, "Mean curvature regularization-based Poisson image restoration," *J. Electronic Imaging*, vol. 24, no. 3, p. 033025, 2015.

[25] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1, pp. 259–268, 1992.

[26] J. Shen, S. H. Kang, and T. F. Chan, "Euler's elastica and curvature-based inpainting," *SIAM J. Appl. Math.*, vol. 63, no. 2, pp. 564–592, 2003.

[27] L. S. Velimirovic, M. S. Ciric, and M. D. Cvetkovic, "Change of the willmore energy under infinitesimal bending of membranes," *Comput. Math. Appl.*, vol. 59, no. 12, pp. 3679–3686, June 2010.

[28] F. Yang, K. Chen, B. Yu, and D. Fang, "A relaxed fixed point method for a mean curvature-based denoising model," *Optim. Method Softw.*, vol. 29, no. 2, pp. 274–285, 2014.

[29] A. Yezzi, "Modified curvature motion for image smoothing and enhancement," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 345–352, Mar 1998.

[30] J. Zhang, K. Chen, F. Chen, and B. Yu, "An efficient numerical method for mean curvature-based image registration model," *East Asian Journal on Applied Mathematics*, vol. 7, no. 1, pp. 125–142, 2017.

[31] H. Zhu, H. Shu, J. Zhou, X. Bao, and L. Luo, "Bayesian algorithms for PET image reconstruction with mean curvature and Gauss curvature diffusion regularizations," *Computers in Biology and Medicine*, vol. 37, no. 6, pp. 793–804, 2007.

[32] W. Zhu and T. Chan, "Image denoising using mean curvature of image surface," *SIAM Journal on Imaging Sciences*, vol. 5, no. 1, pp. 1–32, 2012.

[33] W. Zhu, X.-C. Tai, and T. Chan, "Augmented Lagrangian method for a mean curvature based image denoising model," *Inverse Probl. Imaging*, vol. 7, no. 4, pp. 1409–1432, 2013.

[34] J. Zimmerberg and M. M. Kozlov, "How proteins produce cellular membrane curvature," *Nat. Rev. Mol. Cell Biol.*, vol. 7, no. 1, pp. 9–19, 2006.