LOCAL WEIGHTED GAUSSIAN CURVATURE FOR IMAGE PROCESSING

Yuanhao Gong, Ivo F. Sbalzarini

MOSAIC Group, Center of Systems Biology,
Max Planck Institute of Molecular Cell Biology and Genetics,
Pfotenhauerstr. 108, D–01307 Dresden, Germany.

ABSTRACT

We present a variational model with local weighted Gaussian curvature as regularizer. We show its convexity for an area-weight function and provide a closed-form solution for this case. The corresponding regularization coefficient has a theoretical bound. Moreover, we prove that the model is convex for a wide range of weight functions and show that it can be efficiently solved using splitting techniques. Finally, we demonstrate several applications of the model in image denoising, smoothing, texture decomposition, image sharpening, and regularization-coefficient optimization.

***Index Terms*—** Gaussian curvature, regularization, convex model, variational form, image processing

1. INTRODUCTION

Reconstructing a signal from discrete samples, such as image pixels or a point cloud, is a fundamental task. However, since both the topology and the metric on the samples are missing, it is not clear what the true signal should be, especially in regions devoid of samples. Conceptually, there are two approaches to recovering the signal: interpolation (find missing data) and model fitting (reduce error). Both approaches require predefined basis functions that ideally reflect geometric properties of the signal, such as connectivity, smoothness, sparsity, or curvature. These implicitly assumed properties constitute the prior knowledge about the signal. Their imposition may render the reconstruction problem well-posed. Frequently used priors include sparsity in the spatial and/or frequency domain, total variation (TV), mean curvature (MC) [1, 2, 3], and Gaussian curvature (GC) [4, 5, 2, 6, 7].

Variational methods have been successfully used in image restoration [8, 9, 10, 2, 11], segmentation [12, 13], and inpainting [14]. Here, we show how to impose weighted Gaussian curvature (WGC) priors in a variational framework.

1.1. Variational Framework

Let $S = \{s_i(\vec{x}) : i = 1 \dots N\}$ be the samples with spatial positions $\vec{x} = (x, y)^T$. We aim at recovering an image $U(\vec{x})$ such that

$$\min_{U \in F_s} E(U) = \int_{\vec{x}\in\Omega} \Phi_1(U, S)\,\mathrm{d}\vec{x} + \lambda \int_{\vec{x}\in\Omega} \Phi_2(U)\,\mathrm{d}\vec{x}. \quad (1)$$

Φ_1 is a data-fitting (loss) functional between the recovered image U and samples S. Φ_2 is a regularization functional on U. The parameter λ is a scalar regularization coefficient. Ω is the image domain, and F_s is a suitable function space for U.

Frequently, Φ_1 is a distance metric, such as the Euclidean distance, Mahalanobis distance, Hausdorff distance, or L_p distance. The choice of distance metric depends on how the data were obtained, the noise distribution and magnitude, the targeted reconstruction error, and the desired computational efficiency. Common choices are the L_2 distance to filter Gaussian noise or the L_1 distance to filter outliers.

Φ_2 has to be designed with several goals in mind: 1) it should be efficient to compute; 2) it should have a mathematical meaning; 3) it should generate satisfactory results; 4) it should be easily adopted into different models. Even though there are many well-known regularization terms, such as Tikhonov, the ℓ_2 norm of the gradient, TV, MC, total curvature (TC) [13], etc., none of them fulfills all of these characteristics. We show that WGC regularization has all of the above features.

Recently, curvature regularization has been adopted in variational frameworks for various image-processing problems, including inpainting [14], smoothing [15, 13], and segmentation [13, 16].

1.2. Gaussian Curvature

Let $\vec{\Psi} = (\vec{x}, U(\vec{x}))$ be the image surface. We then have the first and second fundamental form:

$$\boldsymbol{F} = \begin{pmatrix} 1 + U_x^2, & U_x U_y \\ U_x U_y, & 1 + U_y^2 \end{pmatrix} \quad (2)$$

$$\boldsymbol{D} = \begin{pmatrix} \vec{\Psi}_{xx} \cdot \vec{n}, & \vec{\Psi}_{xy} \cdot \vec{n} \\ \vec{\Psi}_{yx} \cdot \vec{n}, & \vec{\Psi}_{yy} \cdot \vec{n} \end{pmatrix}, \quad (3)$$

where subscripts denote differentiation with respect to the corresponding variable. The normal vector is given by $\vec{n} =$

Funding: Swiss National Science Foundation, grant CRSII3-132396/1.

$\frac{(-U_x,-U_y,1)}{\sqrt{1+U_x^2+U_y^2}}$. Gaussian curvature (GC) is defined as:

$$G(U(\vec{x})) = \frac{\det(\boldsymbol{D})}{\det(\boldsymbol{F})} = \frac{U_{xx}U_{yy} - U_{xy}^2}{(1+U_x^2+U_y^2)^2}\,. \quad (4)$$

From the right-hand side of Eq. 4, we have:

$$G(U(\vec{x})) = (U_{xx}U_{yy} - U_{xy}^2)/(1+2\|\nabla U\|_2^2 + \|\nabla U\|_2^4)\,, \quad (5)$$

which means that $U_{xx}U_{yy} - U_{xy}^2$ is a good approximation to G when $\|\nabla U\|_2^2$ is small. This inspires our construction of an area-weighted Gaussian curvature regularizer for variational problems.

Previously, GC has been used in several diffusion-based models [4, 2, 6, 7], which are generally based on the geometric flow [4]

$$\frac{\partial}{\partial t}U = \nabla \cdot (\phi(G)\nabla U) \quad (6)$$

with initial condition $U_0 = S$ and proper boundary conditions. The function ϕ is monotonic. This anisotropic diffusion process is similar to the Perona-Malik model [17]. Edge-indicator weights can be used to preserve edges during flow evolution [7]. A comparison of MC, GC, and TV has been done in Refs. [6, 2]. The model we present here is not diffusion-based.

1.3. Motivation and Contributions

GC is an intrinsic property of the surface and is independent of how the surface is embedded in external coordinates. Moreover, surfaces with zero GC can be isometrically mapped onto a plane without distortion. Minimizing GC can hence be seen as making the image surface As Planar As Possible (APAP). The Ricci flow drives the surface toward constant GC by evolving its Riemann metric [18].

Total GC, however, is related to the surface's topology through the Gauss-Bonnet theorem:

Theorem 1 $\int_{\vec{\Psi}} G\,\mathrm{d}\vec{\Psi} + \int_{\partial\vec{\Psi}} G_b\,\mathrm{d}\vec{b} = 2\pi\chi(\vec{\Psi})$,

where G_b is the boundary curvature, $\mathrm{d}\vec{b}$ a length element, and χ the Euler characteristic of $\vec{\Psi}$. Because of this dependence, we minimize WGC instead of total GC.

A second reason is that the WGC model is more general, since different weight functions can be adopted. The resulting model is convex over a wide range of weight functions.

The Euler-Lagrange equation relates the variational framework (Eq. 1) to diffusion models (Eq. 6). For diffusion flows, however, a CFL stability condition has to be satisfied at every iteration, limiting computational performance especially for large images or videos. Convex models can be solved without any CFL limit using solvers such as Primal/Dual methods [19] or split-Bregman methods [20].

Our contributions here are:

1) We propose a new variational model with WGC regularization that is not based on anisotropic diffusion.
2) We prove that our model is convex for a wide range weight functions and present a closed-form solution for an area-weight function.
3) We provide a theoretical bound for the regularization coefficient and analyze a low-rank approximation to our model.
4) We derive an orthogonal basis that enables multi-resolution analysis for images of the same size.
5) We demonstrate several applications of WGC priors in image denoising, sharpening, and cartoon/texture decomposition.

2. WEIGHTED GAUSSIAN CURVATURE

We take $\Phi_1 = \frac{1}{2}(U-S)^2$ and $\Phi_2 = G(U)\theta(\vec{x})$, where $\theta(\vec{x})$ is a weight function. We further take F_s to be the ℓ^2 space. Then, our model is defined as:

$$\min_{U\in\ell^2} E(U) = \int_{\vec{x}\in\Omega} \frac{1}{2}(U-S)^2\,\mathrm{d}\vec{x} + \lambda \int_{\vec{\Psi}} G(U)\theta\,\mathrm{d}\vec{\Psi}\,. \quad (7)$$

2.1. Closed-Form Solution for Area Weights

In principle, the weight function θ can be chosen arbitrarily. Motivated by Eq. 4, we choose $\theta(\vec{x}) = (1+U_x^2+U_y^2)^{\frac{3}{2}}$, which is related to the surface area element $\mathrm{d}\vec{\Psi} = \mathrm{d}\vec{x}\sqrt{1+U_x^2+U_y^2}$. The resulting WGC hence becomes an area-weighted GC. This weight is also the determinant of the Hessian matrix, which is commonly used for point or line detection, and is also related to tensor diffusion. We hence have:

$$\begin{aligned} G(U)\theta(\vec{x})\mathrm{d}\vec{\Psi} &= \det(\boldsymbol{D})\det(\boldsymbol{F})\mathrm{d}\vec{x} \\ &= (U_{xx}U_{yy} - U_{xy}^2)\mathrm{d}\vec{x}\,. \end{aligned} \quad (8)$$

The resulting energy functional is:

$$E(U) = \int_\Omega \frac{1}{2}(U-S)^2\,\mathrm{d}\vec{x} + \lambda \int_\Omega (U_{xx}U_{yy} - U_{xy}^2)\,\mathrm{d}\vec{x}\,. \quad (9)$$

It can be rewritten in discrete form as:

$$E(\hat{U}) = \frac{1}{2}(\hat{U}-\hat{S})^T(\hat{U}-\hat{S}) + \lambda \hat{U}^T \boldsymbol{W}\hat{U}\,, \quad (10)$$

where $\hat{U}$ and $\hat{S}$ are discrete forms of U and S, respectively. $\boldsymbol{W} = \boldsymbol{A}_{xx}^T\boldsymbol{A}_{yy} - \boldsymbol{A}_{xy}^T\boldsymbol{A}_{xy}$, where $\boldsymbol{A}_{..}$ is the matrix of central-difference approximations to the second derivatives with respect to the subscript variables.

This model has a closed-form solution, which is not possible for diffusion-based models [4, 2, 6, 7]:

$$\frac{\partial}{\partial \hat{U}}E(\hat{U}) = (\hat{U}-\hat{S}) + \lambda \boldsymbol{W}_2\hat{U} = 0 \quad (11)$$

$$\Longrightarrow (\boldsymbol{I} + \lambda\boldsymbol{W}_2)\hat{U} = \hat{S}\,, \quad (12)$$

where $\boldsymbol{W}_2 = \boldsymbol{W}^T + \boldsymbol{W}$.

2.1.1. *Computational Efficiency*

It is worth noting that $\boldsymbol{M} = \boldsymbol{I} + \lambda \boldsymbol{W}_2$ is independent of $\hat{U}$, which means that the entire matrix can be pre-computed and reused for images of the same size, as in a video sequence. Moreover, $\boldsymbol{W}$ is a very sparse matrix, which renders solving the above equation efficient. Table 1 compares the runtime of our model with that of a TV model [8]. This efficiency also allows optimizing the regularization parameter λ using line search.

Image Size	64×64	128×128	256×256
Our Model	0.02496	0.1314	0.5963
TV[1]	2.187	2.354	6.041
TV[2]	0.9409	2.892	5.983

Table 1. Runtime in seconds on a 2 GHz Intel Core i7 using Matlab R2012b.

2.1.2. *Convexity and Bound for* λ

We provide a proof that $\boldsymbol{W}$ is positive-semidefinite (PSD), implying that our model is convex, if λ is bounded. Let

$$\boldsymbol{P} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & \ddots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{pmatrix}$$

be a circulant square matrix of size $mn \times mn$ ($m \times n$ are the image dimensions). It is then clear that $\boldsymbol{P}^{mn} = \boldsymbol{I}$ and $\boldsymbol{A}_{xx} = -2\boldsymbol{I} + \boldsymbol{P} + \boldsymbol{P}^{-1}$. Similarly, $\boldsymbol{A}_{yy} = -2\boldsymbol{I} + \boldsymbol{P}^{m} + \boldsymbol{P}^{-m}$ and $\boldsymbol{A}_{xy} = \frac{1}{4}(\boldsymbol{P}^{m+1} + \boldsymbol{P}^{-m-1} - \boldsymbol{P}^{m-1} - \boldsymbol{P}^{-m+1})$. Therefore, $\boldsymbol{W}$ is PSD with spectral radius $\rho(\boldsymbol{W}) \leq 16$. This implies that $\boldsymbol{M}$ is PSD if $\lambda > -\frac{1}{2\rho(\boldsymbol{W})}$.

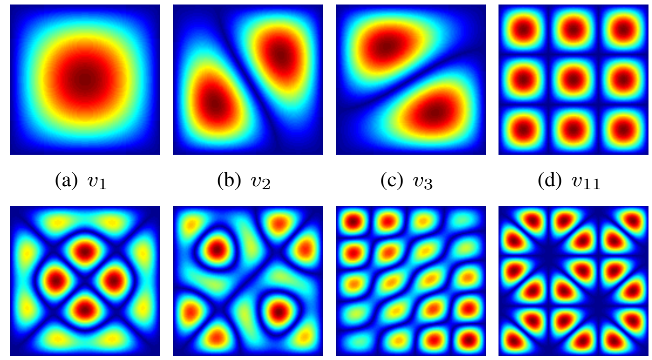
2.1.3. *Multi-Scale Analysis and Low-Rank Approximation*

The eigenvectors v_i of $\boldsymbol{W}$ provide a complete orthogonal basis for all images of size $m \times n$. The resulting basis is reminiscent of certain wavelet filters. Some example basis functions are shown in Fig. 1. They also provide a novel way to solve Eq. 12 in the sense of a low-rank approximation.

Let $\boldsymbol{W} v_i = \gamma_i v_i, i = 1, \dots, (mn)$, where γ_i are the corresponding eigenvalues. Then, $\hat{S} = \sum_{i=1}^{mn} \beta_i v_i$ and $\hat{U} = \sum_{i=1}^{mn} \alpha_i v_i$. Due to orthogonality, $\alpha_i = \frac{\beta_i}{1+2\lambda\gamma_i}$. We solve this linear system for $i < K$, where K is a pre-defined rank-approximation order.

[1] Lena image, $\epsilon = 1$, max iteration=80

[2] cameraman image, $\epsilon = 1$, max iteration=80



(a) v_1 (b) v_2 (c) v_3 (d) v_{11}

(e) v_{16} (f) v_{17} (g) v_{24} (h) v_{29}

Fig. 1. Some example basis functions v_i.

2.2. General Weights with Edge Preservation

In the general case when $\theta(\vec{x}) = \omega(\|\nabla\vec{\Psi}\|)$, Eq. 7 can be rewritten in discrete form as:

$$E(\hat{U}) = \frac{1}{2}(\hat{U} - \hat{S})^T(\hat{U} - \hat{S}) + \lambda \hat{U}^T \boldsymbol{W}\boldsymbol{W}_\omega \hat{U})\,, \qquad (13)$$

where $\boldsymbol{W}_\omega$ is a diagonal matrix corresponding to the weight function ω. More specifically, $\boldsymbol{W}_\omega(i,i) = \frac{\omega(\|\nabla\vec{\Psi}\|)}{(1+U_x^2+U_y^2)^{\frac{3}{2}}}$. Then:

$$\frac{\partial}{\partial \hat{U}} E(\hat{U}) \approx (\hat{U} - \hat{S}) + \lambda \boldsymbol{W}_2 \boldsymbol{W}_\omega \hat{U} = 0 \qquad (14)$$

$$\Longrightarrow (\boldsymbol{I} + \lambda \boldsymbol{W}_2 \boldsymbol{W}_\omega)\hat{U} = \hat{S}\,. \qquad (15)$$

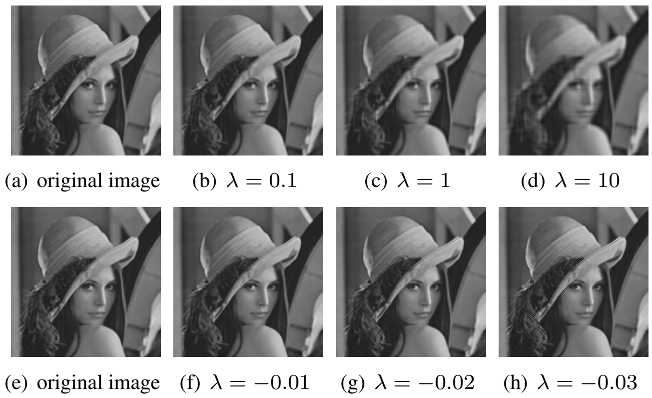
This model has no closed-form solution. However, it can be efficiently solved using the split weighted Gaussian curvature (SWGC) algorithm given below.

Algorithm 1 SWGC: Split Weighted Gaussian Curvature

Require: $\hat{S}$, λ
compute $\boldsymbol{W}_2$, $\boldsymbol{W}_\omega(\hat{S})$, set $\hat{U}_0 = \hat{S}$, $k = 0$
while $\max\{|\hat{U}_{k+1} - \hat{U}_k|\} < tol$ **do**
  compute $\boldsymbol{W}_{\omega(\|\nabla\vec{\Psi}\|)}$
  compute $\hat{U}_{k+1}$ from Eq. 15
  $k = k + 1$
end while
Ensure: $\hat{U}$

2.2.1. *Convexity and Bound for* λ

$\boldsymbol{W}_\omega$ is PSD for a wide range of functions ω. Therefore, $\boldsymbol{W}\boldsymbol{W}_\omega$ is PSD and λ is bounded as $\lambda > -\frac{1}{2\rho(\boldsymbol{W}\boldsymbol{W}_\omega)}$. It is straightforward to get a bound for ω. For example, when ω is the identity function, $\rho(\boldsymbol{W}_\omega) \leq 1$.



(a) original image (b) $\lambda = 0.1$ (c) $\lambda = 1$ (d) $\lambda = 10$

(e) original image (f) $\lambda = -0.01$ (g) $\lambda = -0.02$ (h) $\lambda = -0.03$

Fig. 2. Image smoothing (top row) and sharpening (bottom row) using area-weighted GC for different λ.

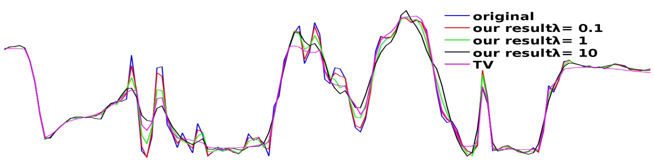


Fig. 3. A line profile from Fig. 2 for image smoothing.

3. APPLICATIONS

We demonstrate the application of WGC in image smoothing, sharpening, cartoon/texture decomposition, image denoising, and regularization-coefficient optimization.

3.1. Image Smoothing and Sharpening

Figure 2 shows the results of image smoothing and sharpening using our WGC model with the area-weight function and with different parameters (sharpening with negative $\lambda > -\frac{1}{2\rho(\boldsymbol{W})}$). A line profile is compared with TV[3] in Fig. 3.

In Fig. 4, image smoothing is shown with edge-preserving weights (ω is the identity function); a detail patch is shown in the row below. In practice, three to four iterations of SWGC are enough.

3.2. Cartoon/Texture Decomposition and Denoising

We compare the area-weighted GC model with TV regularization for cartoon/texture decomposition and image denoising. The result is shown in Fig. 5 for $\epsilon = 1.5$ in TV and $\lambda = 30$ in WGC.

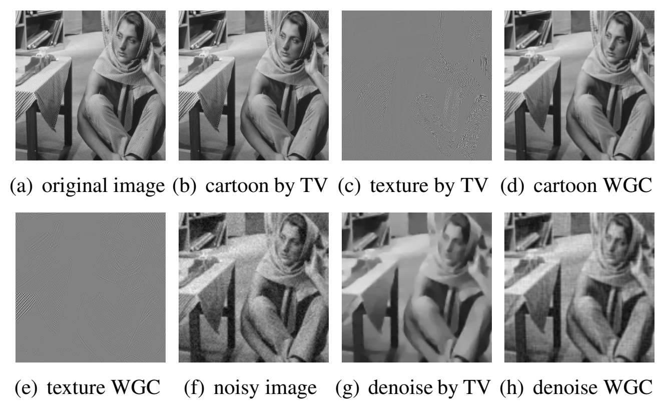
For denoising, the image is corrupted with additive Gaussian noise of magnitude $\sigma = 10$. Denoising results by TV[3] and WGC (area weight) with $\lambda = 0.48$ are shown in Fig. 5. The final mean-square errors for TV and WGC are 70.77 and 56.54, respectively.

[3] TV uses $\epsilon = 1$ and max iteration = 80

(a) original image (b) $\lambda = 0.1$ (c) $\lambda = 1$ (d) $\lambda = 10$

(e) original image (f) $\lambda = 0.1$ (g) $\lambda = 1$ (h) $\lambda = 10$

Fig. 4. Image smoothing with edge-preserving weights (ω is the identity function).



(a) original image (b) cartoon by TV (c) texture by TV (d) cartoon WGC

(e) texture WGC (f) noisy image (g) denoise by TV (h) denoise WGC

Fig. 5. Cartoon/texture decomposition and denoising.

3.3. Regularization-Coefficient Optimization

To the best of our knowledge, searching for the optimal regularization coefficient λ is hard in general. The computational efficiency of our model, however, allows the use of line search to optimize the regularization coefficient in Eq. 7 by solving Eq. 12. We did this for the image denoising experiment above.

4. CONCLUSION AND FURTHER WORK

We have presented weighted Gaussian curvature regularization in a variational framework. The resulting model is convex over a wide range of weight functions and has a closed-form solution for the special case of area weights. We have shown a bound on λ and presented an efficient algorithm to numerically solve the model when no closed-form solution is available. We have demonstrated the proposed model in several applications ranging from image smoothing to sharpening, denoising, and cartoon/texture decomposition.

Weighted Gaussian curvature can be further extended to 3D images and to point-cloud surfaces [21].

5. REFERENCES

[1] Adel I. El-Fallah and Gary E. Ford, "Mean curvature evolution and surface area scaling in image filtering," *IEEE Trans. Image Proc.*, vol. 6, no. 5, pp. 750–753, 1997.

[2] 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.

[3] Xilin Liu, Zhengwei Ying, and Shufang Qiu, "A fourth-order partial differential equations method of noise removal," in *Image and Signal Processing (CISP), 2011 4th International Congress on*, oct. 2011, vol. 2, pp. 641–645.

[4] Suk-Ho Lee and Jin Keun Seo, "Noise removal with Gauss curvature-driven diffusion," *IEEE Trans. Image Proc.*, vol. 14, no. 7, pp. 904–909, 2005.

[5] Huanxi Zhao and Guoliang Xu, "Triangular surface mesh fairing via Gaussian curvature flow," *J. Comput. & Appl. Math.*, vol. 195, no. 1, pp. 300–311, 2006.

[6] Niels Overgaard and Jan Solem, *The Variational Origin of Motion by Gaussian Curvature*, vol. 4485 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 2007.

[7] Bibo Lu, Hui Wang, and Zhonghua Lin, "High order Gaussian curvature flow for image smoothing," in *Multimedia Technology (ICMT), 2011 International Conference on*, july 2011, pp. 5888–5891.

[8] Leonid I. Rudin, Stanley Osher, and Emad Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1, pp. 259–268, 1992.

[9] Tony Chan, Antonio Marquina, and Pep Mulet, "High-order total variation-based image restoration," *SIAM J. Sci. Comput.*, vol. 22, no. 2, pp. 503–516, 2000.

[10] T. Chan, S. Esedoglu, F. Park, and A. Yip, "Total variation image restoration: Overview and recent developments," *Handbook of mathematical models in computer vision*, pp. 17–31, 2006.

[11] Giannis Chantas, Nikolaos P. Galatsanos, Rafael Molina, and Aggelos K. Katsaggelos, "Variational Bayesian image restoration with a product of spatially weighted total variation image priors," *IEEE Trans. Image Proc.*, vol. 19, no. 2, pp. 351–362, 2010.

[12] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Proc.*, vol. 10, no. 2, pp. 266–277, feb 2001.

[13] B. Goldluecke and D. Cremers, "Introducing total curvature for image processing," in *ICCV 2011*. IEEE, 2011, pp. 1267–1274.

[14] Jianhong Shen, Sung Ha Kang, and Tony F. Chan, "Euler's elastica and curvature-based inpainting," *SIAM J. Appl. Math.*, vol. 63, no. 2, pp. 564–592, 2003.

[15] D. Tschumperlé, "Fast anisotropic smoothing of multi-valued images using curvature-preserving pde's," *Intl. J. Comp. Vis.*, vol. 68, no. 1, pp. 65–82, 2006.

[16] T. Schoenemann and D. Cremers, "Introducing curvature into globally optimal image segmentation: Minimum ratio cycles on product graphs," in *ICCV 2007*. IEEE, 2007, pp. 1–6.

[17] Pietro Perona and Jitendra Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. PAMI*, vol. 12, no. 7, pp. 629–639, 1990.

[18] X. Gu, S. Wang, J. Kim, Y. Zeng, Y. Wang, H. Qin, and D. Samaras, "Ricci flow for 3d shape analysis," in *ICCV 2007*. IEEE, 2007, pp. 1–8.

[19] Mingqiang Zhu, Stephen J. Wright, and Tony F. Chan, "Duality-based algorithms for total-variation-regularized image restoration," *Comput. Opt. & Appl.*, vol. 47, no. 3, pp. 377–400, 2010.

[20] T. Goldstein and S. Osher, "The split Bregman method for L1-regularized problems," *SIAM J. Imaging Sci.*, vol. 2, no. 2, pp. 323–343, 2009.

[21] Y. Gong, G. Paul, and I. F. Sbalzarini, "Coupled signed-distance functions for implicit surface reconstruction," in *9th IEEE International Symposium on Biomedical Imaging (ISBI)*, May 2012, pp. 1000–1003.