CVPR SEATTLE WASHINGTON JUNE 16-18 2020

Minimizing Discrete Total Curvature for Image Processing

Qiuxiang Zhong, Yutong Li, Yijie Yang and Yuping Duan*

Center for Applied Mathematics, Tianjin University, China

Motivation & Strategy

- **Motivation:** **The curvature regularities have the advantage of providing strong priors in the continuity of edges in image processing. Owing to the highly nonlinear, nonsmooth and nonconvex properties of the curvature regularities, the numerical solution are challenging.**
- **Strategy:** **We put forward the novel total curvature for image processing tasks, which has the properties, i.e.,**
  - ***Ideally recover the corners and edges of images***
  - ***Without solving any high order PDE and high efficiency***
  - ***Flexible to adapt with different tasks in image processing***

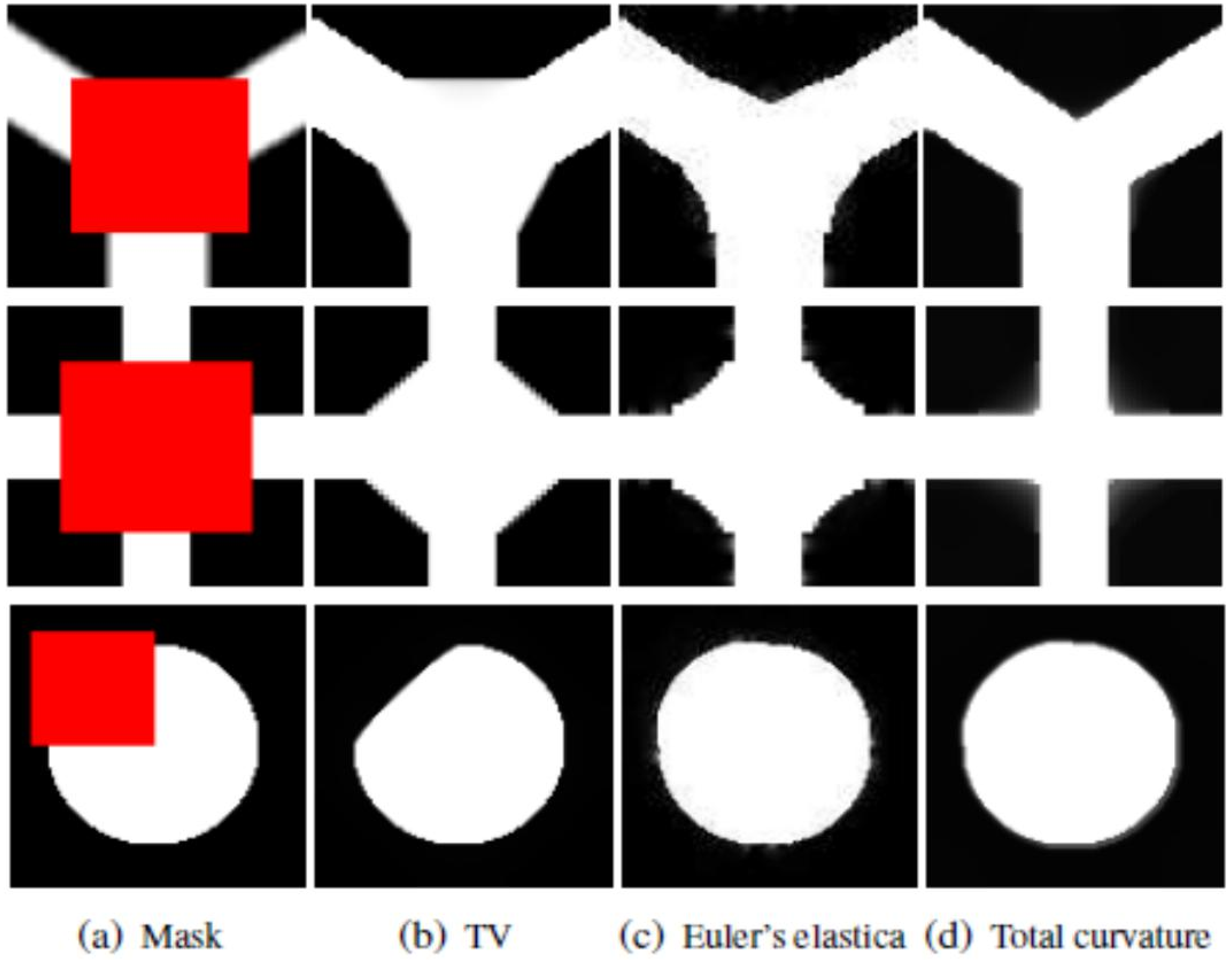


Figure 1. Total curvature is an ideal regularity for image inpainting.

Model

➢ Energy function

- **We concern with the following curvature-based energy minimization problem**

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

- **Here $\boldsymbol{\phi}(\boldsymbol{\kappa})$ is some suitable functions of the curvature, e.g.,**

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

- **The data fidelity term $\mathcal{D}(\boldsymbol{u}, \boldsymbol{f})$ varies according to different image processing tasks such as**

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

Normal Curvature

- **Normal curvature can be expressed as the quotient of the second fundamental form $\mathbf{II}$ and the first fundamental form $\mathbf{I}$ of the surface:**

$$\kappa_n = \frac{\mathbf{II}}{\mathbf{I}} \approx \frac{2d}{ds^2}$$

- **The first fundamental form $\mathbf{I}$ can be estimated using the arc length, i.e., $\mathrm{I} = ds^2$.**
- **The second fundamental form $\mathbf{II}$ can be approximated by $\mathrm{II} \approx 2d$, where d indicates the projection distance of its neighboring point to the tangent plane.**
- **Assume T_{XYZ} be the tangent plane of O. We compute the normal curvature as follows.**
- **We use the half point H to estimate the projection distance d to the tangent T_{XYZ}**

$$d = \frac{2u_{i,j} - u_{i,j-1} - u_{i,j+1}}{2\sqrt{(2u_{i-1,j} - u_{i,j-1} - u_{i,j+1})^2 + (u_{i,j-1} - u_{i,j+1})^2 + 4}}$$

- **The arc length $\widehat{OH}$ can be approximated by**

$$ds = \widehat{OH} \approx \sqrt{(u_H - u_O)^2 + h^2}$$

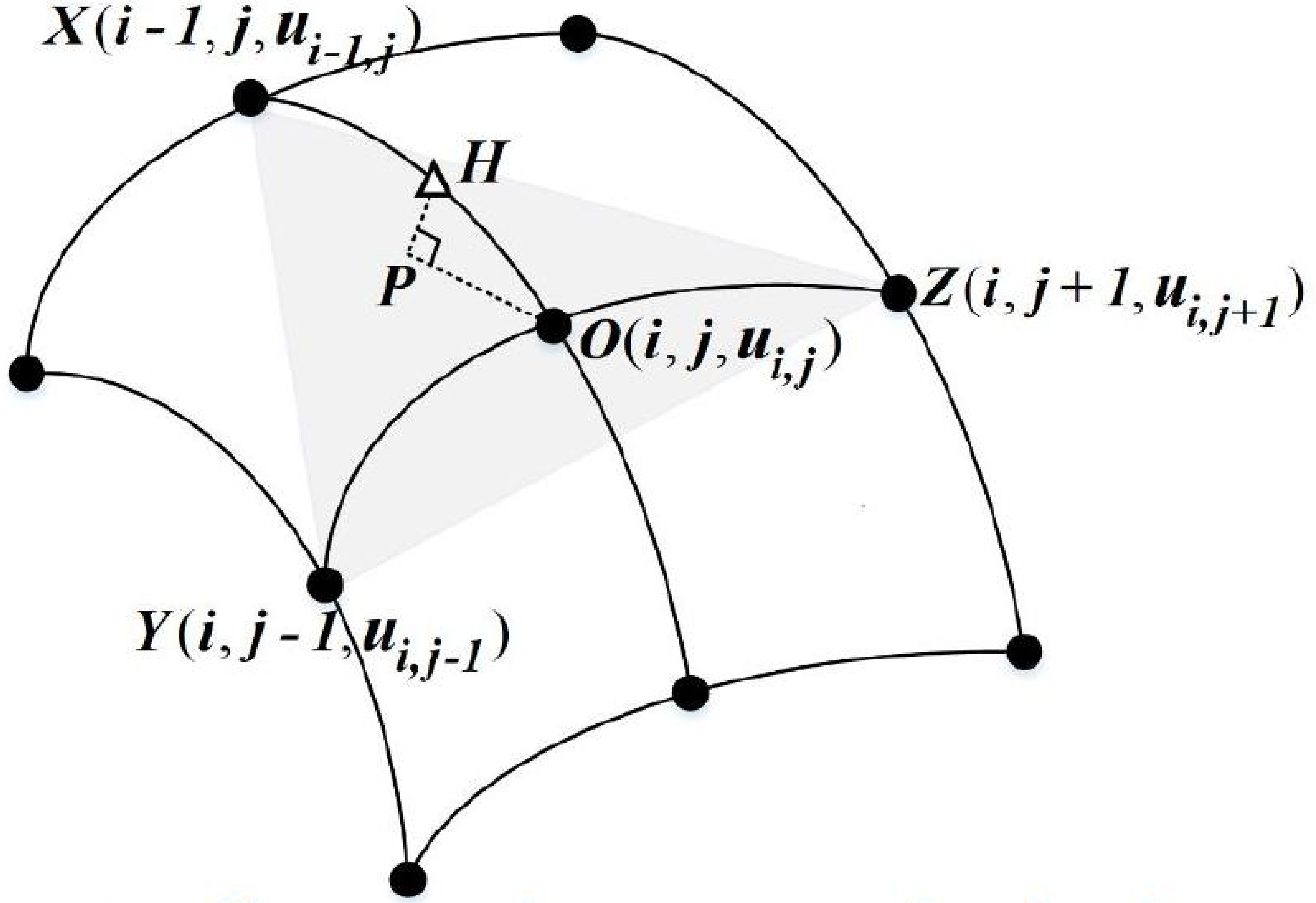


Figure 2. Illustration on calculation of normal curvature.

Total Curvature

➢ Total curvature

- **We estimate eight different normal curvatures in the 3×3 local window over each point on image surface.**

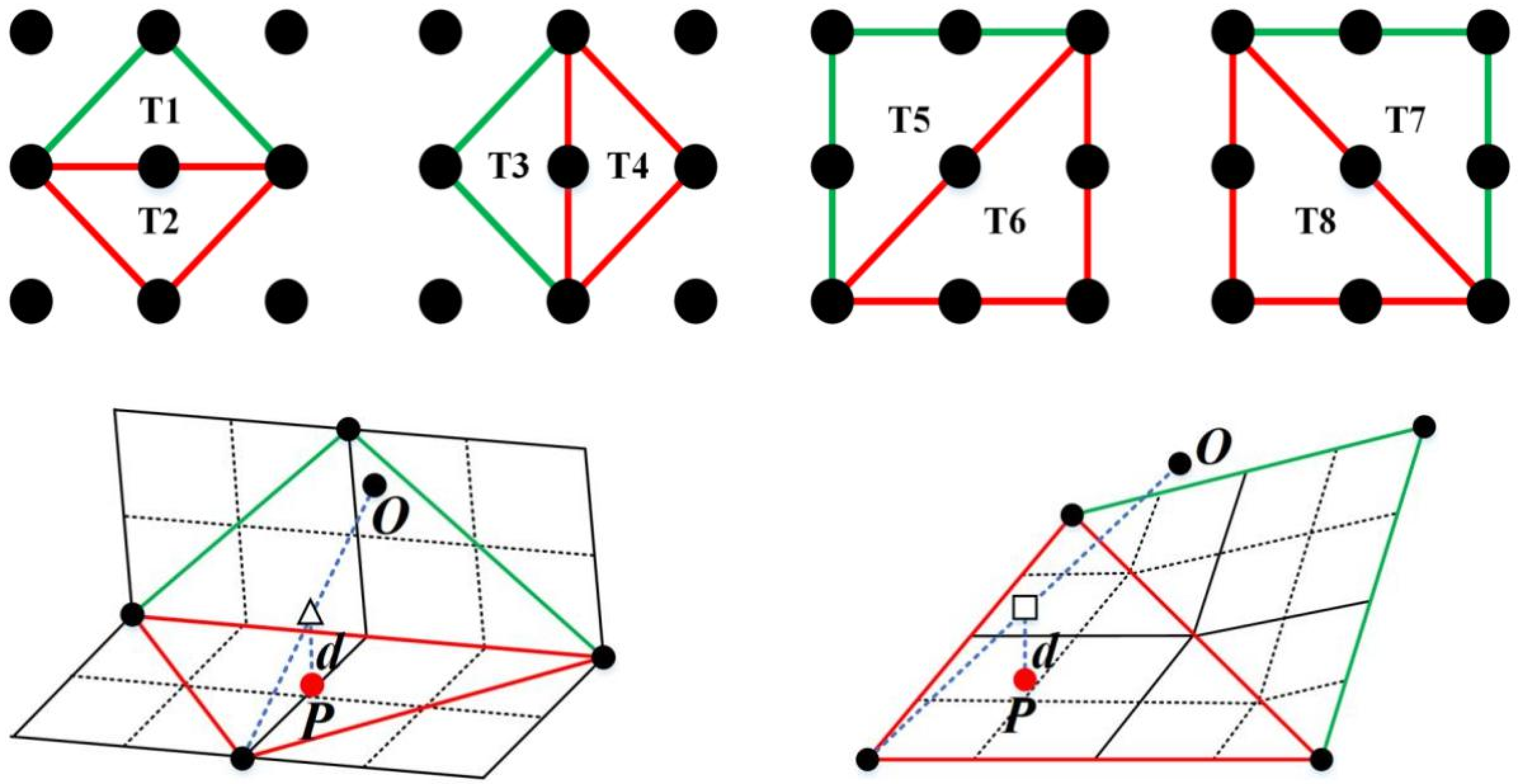


Figure 3. The eight tangent planes of the center point in a 3×3 local window and the representatives of the two different kinds of projections.

- **As a result, the eight normal curvatures can be calculated over each point:**

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

- **Eventually, we introduce the total curvature as a geometric measurement of image:**

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

Algorithm

➢ ADMM-based algorithm

- **We introduce an auxiliary variable $\boldsymbol{v}$ to formulate the associated augmented Lagrangian functional:**

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

- **Sub-minimization w.r.t. $\boldsymbol{u}$**

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

- **Sub-minimization w.r.t. $\boldsymbol{v}$**

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

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:** $u^0 = f$, $v^0 = 0$, $\Lambda^0 = 0$.
3: **while** (not converged and $k \le T_{max}$) **do**

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

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

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

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

$$v^{k+1} = \arg\min_v \Big\{\int_\Omega \phi(\kappa)|v|dx + < \Lambda^k, v - \nabla u^{k+1} > + \frac{\mu}{2}\|v - \nabla u^{k+1}\|^2\Big\}; \quad (13)$$

(iv) Update Λ^{k+1} from:

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

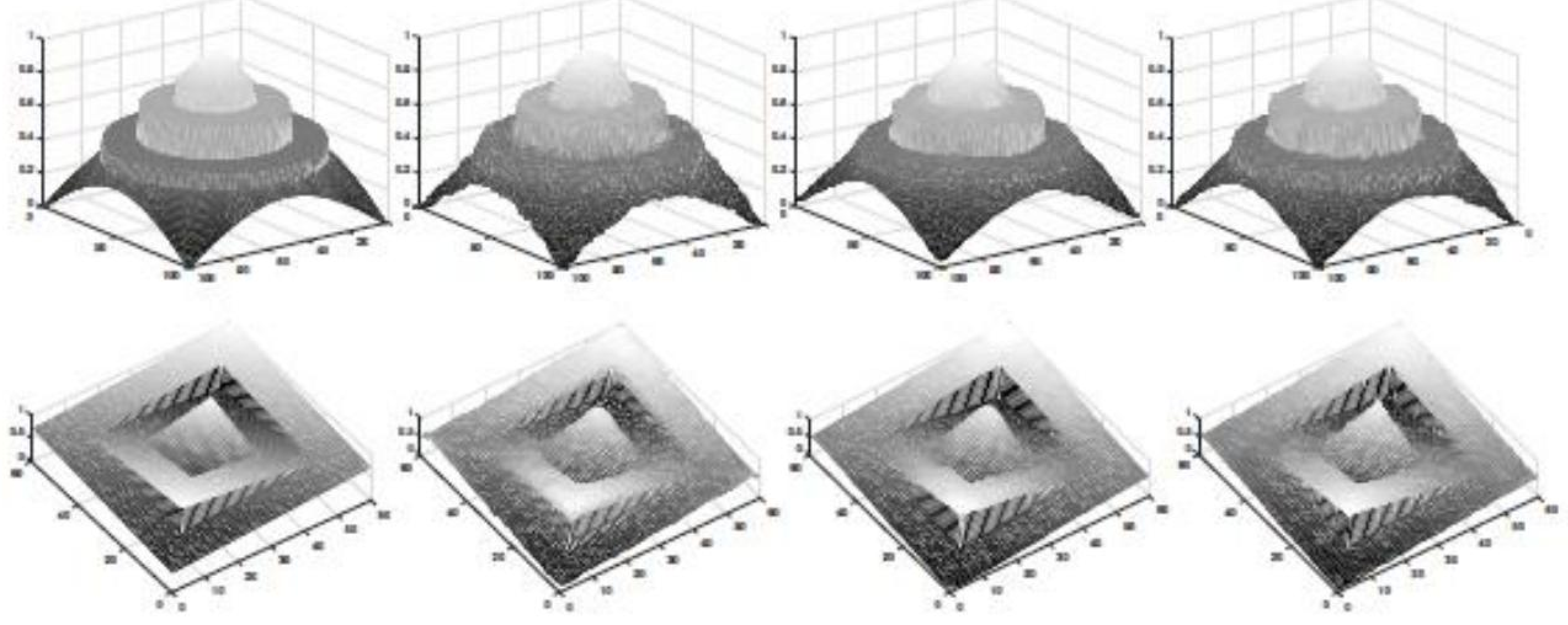
(v) Check convergence condition:

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

4: **end while**

Numerical Experiments

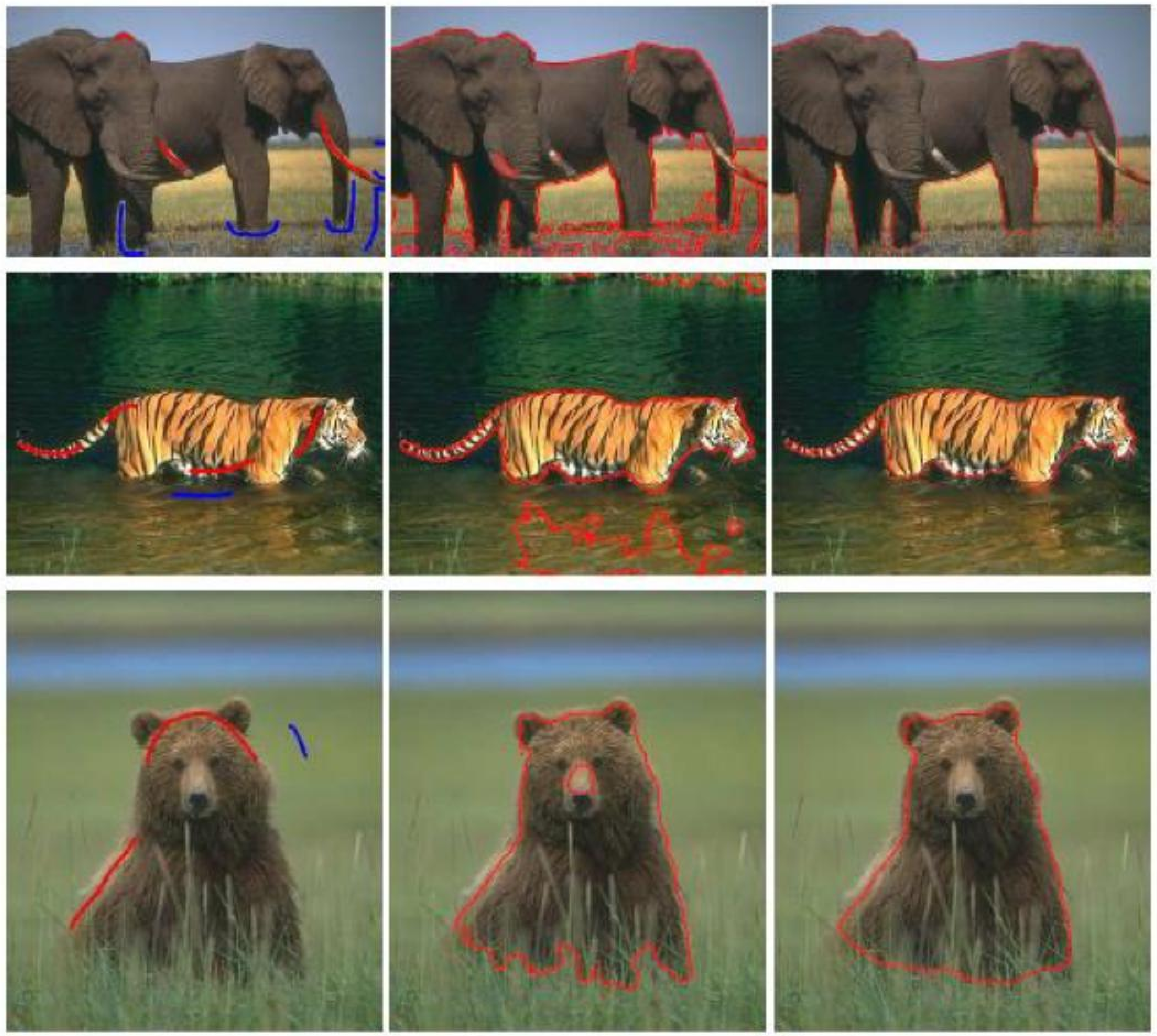
➢ Image denoising



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

Figure 4. 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.

➢ Image segmentation



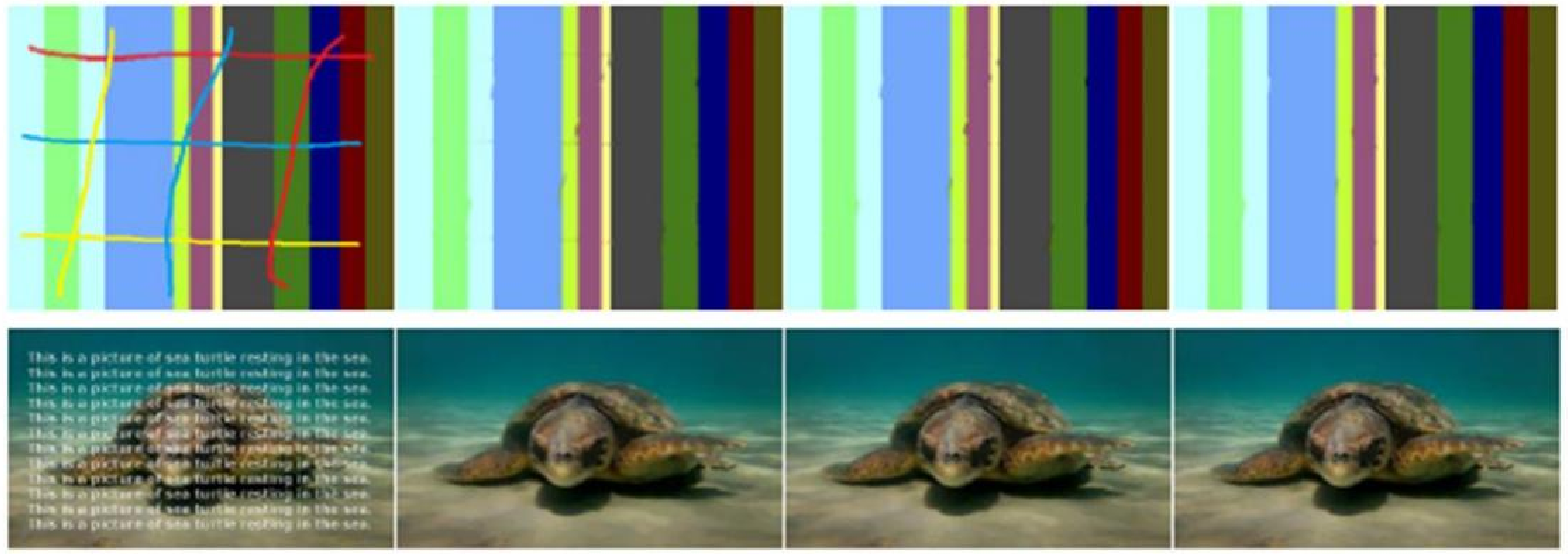
(a) Mask images (b) Euler's elastica (c) TC

Figure 5. Supervised segmentation for natural images.

Image Inpainting

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

Figure 6. The inpainting results of different real images.



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

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

Conclusion

- **We introduced the total curvature regularity for image processing, 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.**

- **The total curvature regularity is applicable to various image processing tasks like denoising, segmentation and inpainting, etc.**

Thank for your attention!

Qiuxiang Zhong, Yutong Li, Yijie Yang and Yuping Duan*
Center for Applied Mathematics, Tianjin University, China