Pacific Graphics 2016
E. Grinspun, B. Bickel, and Y. Dobashi
(Guest Editors)

Volume 35 (2016), Number 7

Piecewise smooth reconstruction of normal vector field on digital data
supplementary material

David Coeurjolly[1] Marion Foare[2] Pierre Gueth[1] Jacques-Olivier Lachaud[2]

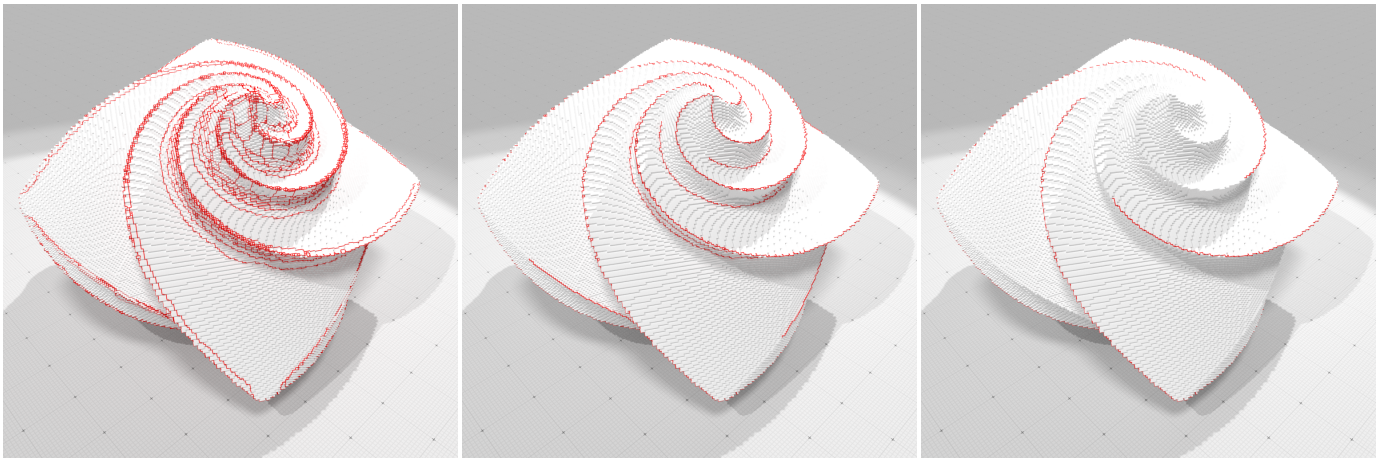
[1]Université de Lyon, CNRS, LIRIS UMR 5205, F-69622, France
[2]Laboratoire de Mathématiques, CNRS, UMR 5127, University Savoie Mont Blanc, France

In this document, we provide additional results of the Ambrosio-Tortorelli approach to reconstruct a piecewise smooth normal vector field preserving edges and features. The following sections also illustrate the role of each parameter (λ, α and ε) on the results (feature or normal vector field).

If not specified, the input normal vector field has been obtained using Integral Invariant (II) estimator with $r = 5$.

1. Scale-space induced by λ parameter

In the AT functional, the λ coefficient has a deep impact on the length of the feature 0-form $\mathbf{v}$. If we fix the other parameters, λ can be interpreted as a scale-space parameter for the feature extraction (see Figure 1). Please note that the inverse of the λ value is proportional to a length.



$\lambda = 0.001$ $\lambda = 0.01$ $\lambda = 0.1$

Figure 1: *Feature extraction on noise free OctaFlower* (256^3) *with* $\alpha = 0.01$ *and* $\varepsilon = [2, 0.25]$.

2. Robustness to noise

The α parameter allows us to control the strength of the regularisation and feature extraction with respect to the amount of noise in the input object. In Figure 2, we present different feature extractions and normal vector fields illustration (normal vector components are mapped to RGB values) for various noise level k and various parameters α (for fixed λ). Even if the feature is quite stable with respect to k, better results can be obtained when α decreases as k increases.

3. Width of features with ε parameter

The ε parameter controls the Γ-convergence of the AT functional. As described in the optimization algorithm, this value decreases to zero and controls the width of the extracted feature (see Figure 3).

4. Stability to input normal estimator

In this section, we quantity the normal vector field estimtion with respect to anisotropic estimators such as VCM and RHT. We also quantify the improvement obtained by AT when such estimators are used as input.

Fig. 4 shows errors between computed normal vector fields and the ground truth versus a range of radii of the estimators for a rotated cube. As this radius increases, we expect the normal vector field to smooth out the edges of the cube. Errors are quantified as the mean angular error (left subfigure) and as the fraction of normal vectors that deviate by more than 5° from the truth. Dotted lines correspond to the input normal vector field $\mathbf{g}$, as solid lines stand for regularized normal vector field $\mathbf{u}$. Digital surfaces are perturbed with same noise as the third row of Fig. 2 ($k = 0.4$). To compensate for the increase of smoothing brought by the increase of estimator radius r, parameter α is computed as $\alpha = 2.5\ 10^{-3}\ r^2$. This ensures a constant smoothing of the regularized normal vector field across the figure.

For small values of r, the estimated normal vector fields suffer from large perturbations as AT regularized fields offer much lower errors. For $r > 6$, differences between estimated and regularized normal vector fields tends to disappear since all the smoothing is done by normal vector estimators. Using the method described in this paper, one can computed normal vector fields for a fraction of estimation radius when compared to direct estimation. Note that normal vector field smoothing is globally independent of feature extraction since λ is constant all across the figure.

Fig. 5 shows the same normal vector field errors as in Fig. 4 for a range of λ values. The radii of the estimators is set to $r = 5$. Parameter λ can be robustly chosen in a wide range spanning an order of magnitude without having a great impact on accuracy ($2.5\ 10^{-3} < \lambda < 5\ 10^{-2}$). Moreover note that this range is independent of the normal vector estimator used for input data.

Figures 6 and 7 show snapshots of rendered regularized normal vector fields for the three estimators for $r = 5$ and $r = 8$ with the same parameters (k, λ and α) as in Fig. 4. For $r = 5$ (middle of the graphs in Fig. 4), AT clearly improves the estimated normal vector field while preserving edges. If we consider $r = 8$ (right part of the graphs in Fig. 4), even if the error measures are similar, we get with AT slight visual improvements in the faces (see RHT for example). As discussed above, $r = 8$ introduces too much smoothness which prevents AT to capture the features.

5. Stability of feature form v

As seen in the previous section, the AT functional is able to localize sharp features even on strongly perturbated data. This is due to the fact that this functional is optimized in a coarse-to-fine way, by decreasing ε progressively. A large ε diffuses features and facilitates the initial guess of their precise position. Decreasing ε thins features down progressively around their best possible localization, which explains the robustness of the approach. This progressive optimization process is illustrated on Fig. 8. Even on a strongly perturbated shape, features are very well captured at a coarse level ($\varepsilon = 2$). Then, as ε is decreased, the AT functional tries at best to localize precisely the sharp features and is helped by this coarse-to-fine process.

6. Comparison of our approach with classical methods on flat domains

In this section, we compare the results of the minimization of AT on an image using finite differences, finite elements with adaptative mesh [BC00] and our method. In this case, g represents the graylevels of an input image, u is a smooth reconstruction of g and v represents the set of discontinuities of g.

With finite differences, Fig. 9(a-b) shows that features are thick ($\{v \approx 0\}$ is approximately 6 pixel wide) and that u is smooth except on the feature strip $\{v \approx 0\}$, where its value is equal to the noisy input data.

Fig. 9(c-d) shows the results obtained by the finite element method of Bourdin *et. al.* in [BC00]. The set of discontinuities v looks thin but since the method adapts and refines the mesh locally around discontinuities, v is still about 5 triangles thick.

Finally, Fig. 9(e) shows the result of AT minimization with our approach. We can see that $\mathbf{v}$ is 1-dimentional at optimum, without changing the domain discretization.

7. Comparison of our approach with a convex relaxation of AT on flat domains

Kee an Kim proposed in [KK14] three convex relaxations of AT : a quadratic relaxation, a linear relaxation, and the last one based on a factorization theorem due to McCormick [McC76]. They used a gradient descend algorithm to compute the minimizers, and observed that the best results are obtained with the linear relaxation.

We compare in Fig. 10 the results of Kee and Kim, extracted from [KK14], and our approach on images they used in their paper.

We can see that the McCormick relaxation (Fig. 10 (a)) retains a lot of artefacts while the quadratic relaxation (Fig. 10 (b)) leads to blurry reconstructions. The results obtained with the linear relaxation (Fig. 10 (c)) are piecewise smooth, but we observe a poor reconstruction of angles in the vectorial case. Furthermore we see on Fig. 10 (d) that they obtain diffuse and even false contours.

References

[BC00] BOURDIN B., CHAMBOLLE A.: Implementation of an adaptive finite-element approximation of the mumford-shah functional. *Numerische Mathematik 85* (2000), 609–646. 2, 8

[KK14] KEE Y., KIM J.: A convex relaxation of the ambrosio–tortorelli elliptic functionals for the mumford-shah functional. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 4074–4081. 2, 3, 9

[McC76] MCCORMICK G. P.: Computability of global solutions to factorable nonconvex programs: Part i - convex underestimating problems. *Mathematical programming 10*, 1 (1976), 147–175. 2

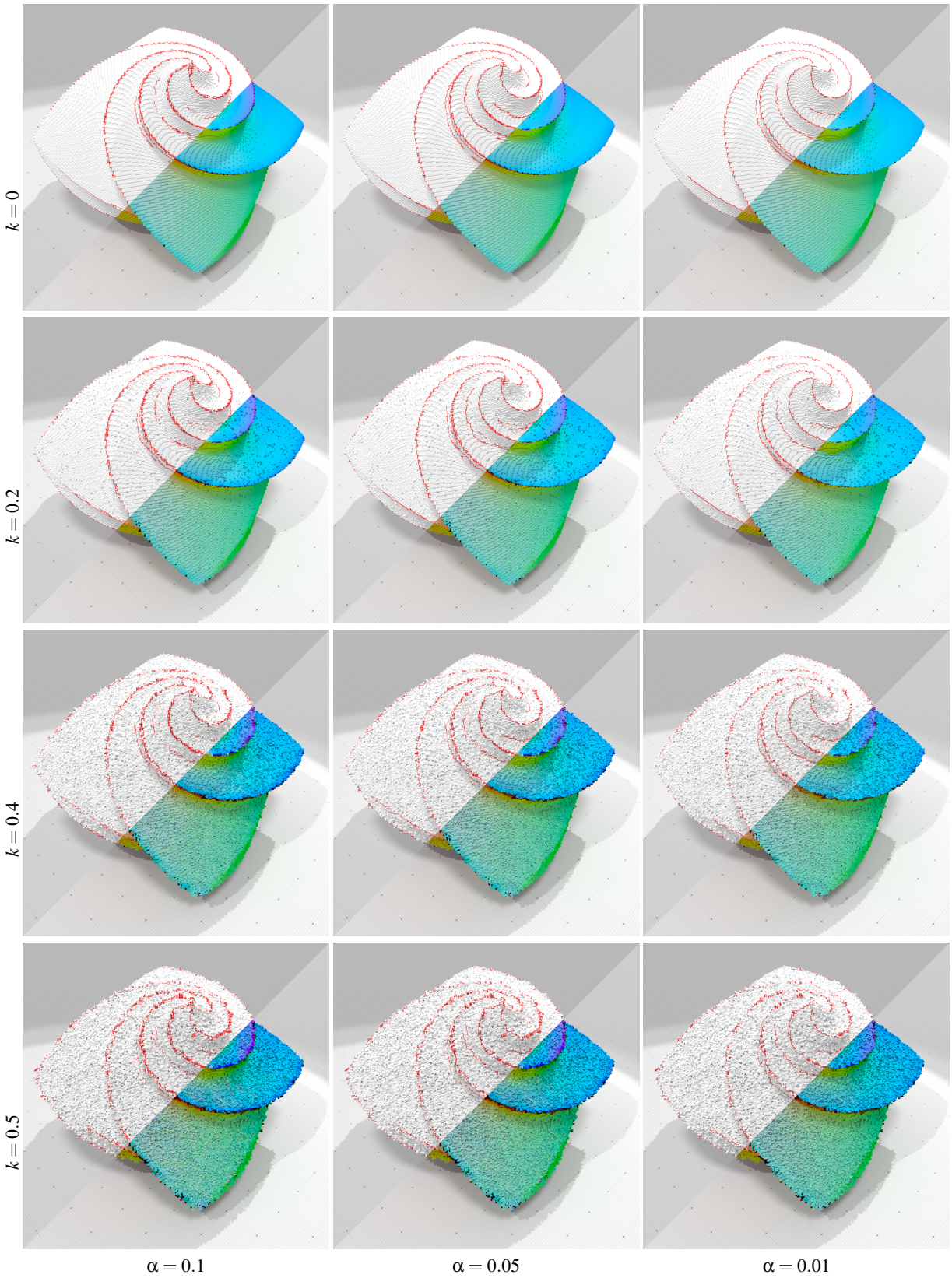


Figure 2: *Feature extraction and normal vector field illustration on noise free OctaFlower* (256^3) *with* $\lambda = 0.01$ *and* $\varepsilon = [2, 0.25]$.

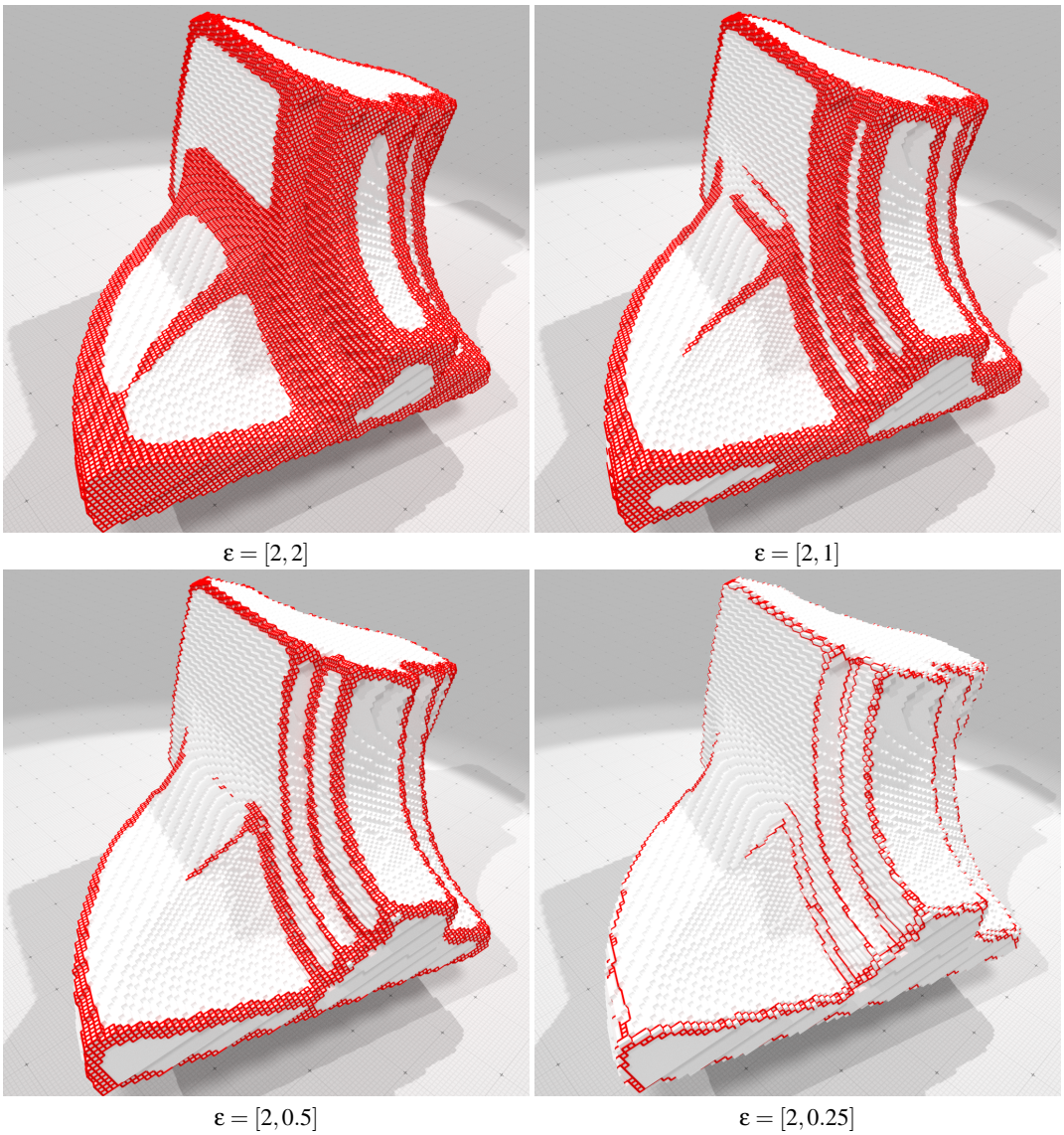


Figure 3: *Feature extraction on a Fandisk* (128^3) *with* $\alpha = 0.05$ *and* $\lambda = 0.01$.

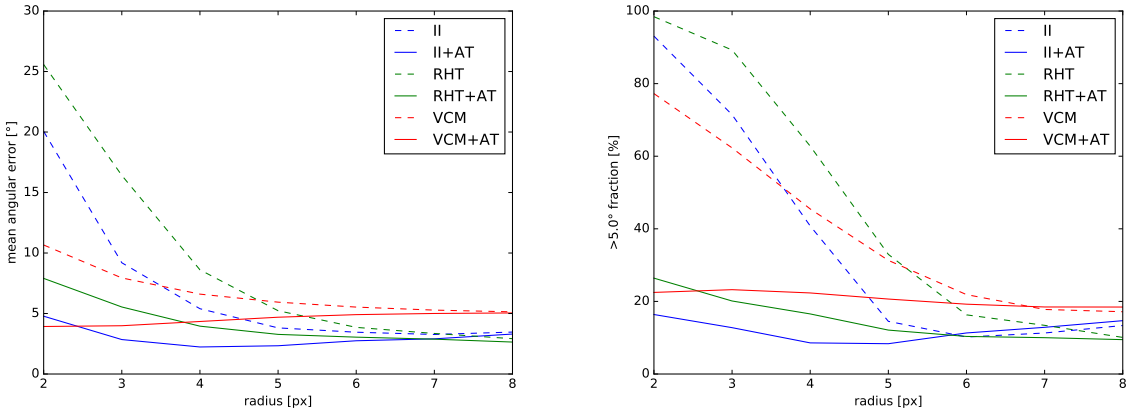


Figure 4: *Estimated and regularized vector fields errors versus estimator radii (and* α*).* $\varepsilon = [2, 0.25]$*, noise* $k = 0.4$*,* $\lambda = 2.5\ 10^{-2}$*,* $\alpha = 2.5\ 10^{-3}\ r^2$.

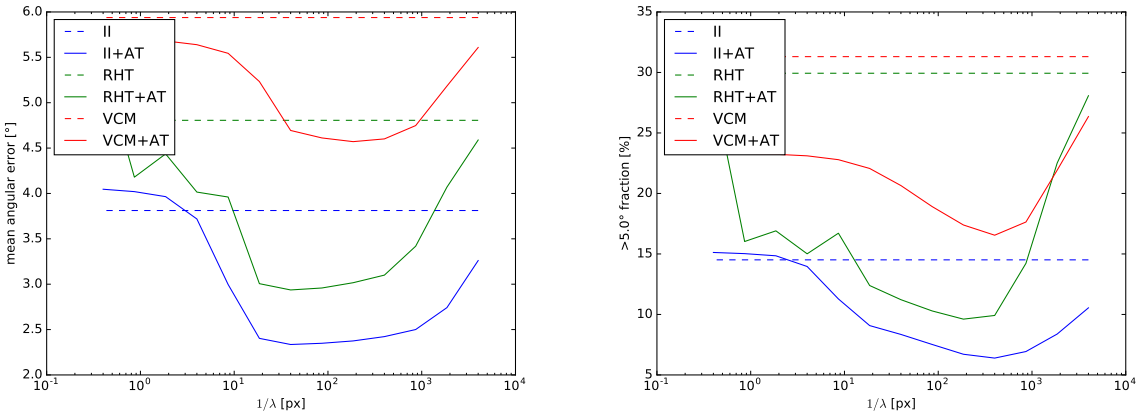


Figure 5: *Estimated and regularized vector fields errors versus prescribed feature perimeter.* $\varepsilon = [2, 0.25]$*, noise* $k = 0.4$*,* $r = 5$*,* $\alpha = 6.25\ 10^{-2}$

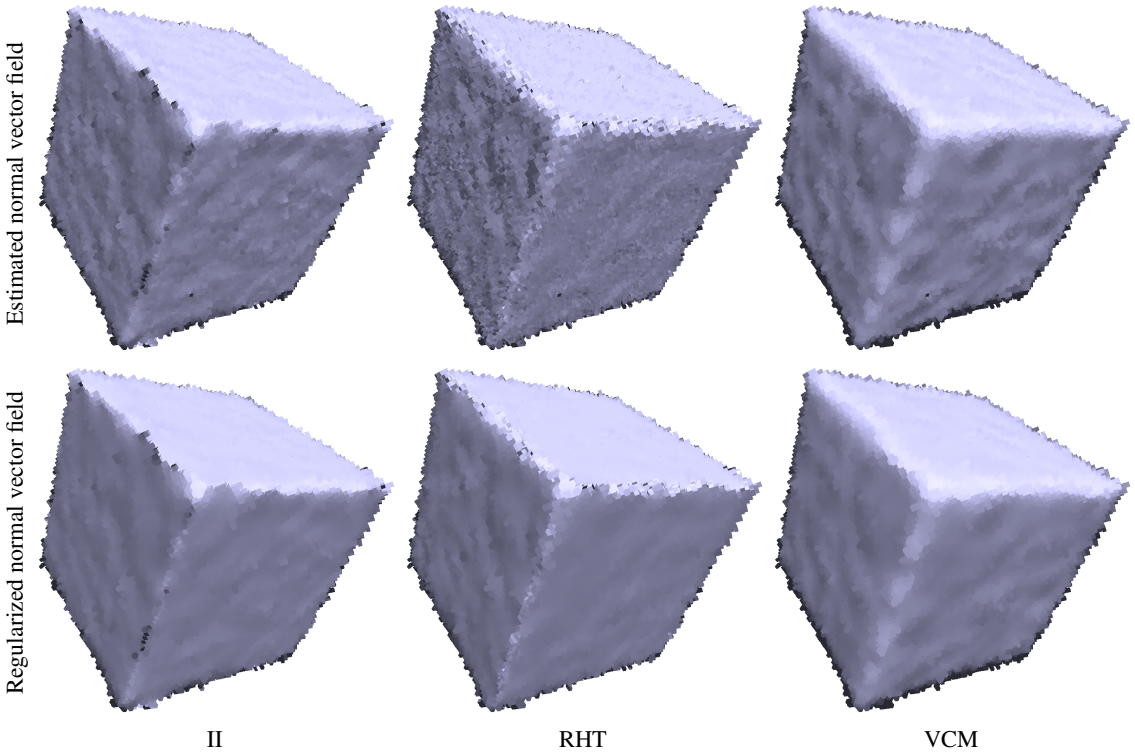


Figure 6: *Estimated and regularized normal vector fields.* $k = 0.4$, $\alpha = 0.16$, $\lambda = 2.5\ 10^{-2}$, $\varepsilon = [2, 0.25]$, $r = 5$.

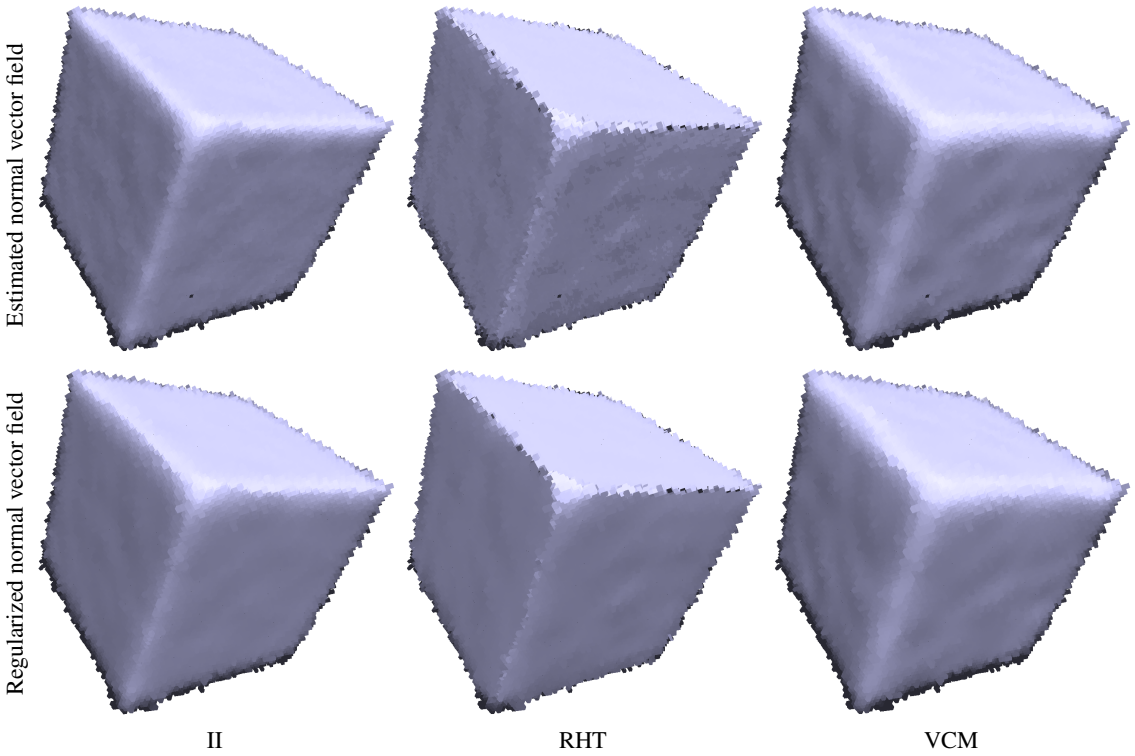


Figure 7: *Estimated and regularized normal vector fields.* $k = 0.4$, $\alpha = 0.16$, $\lambda = 2.5\ 10^{-2}$, $\varepsilon = [2, 0.25]$, $r = 8$.

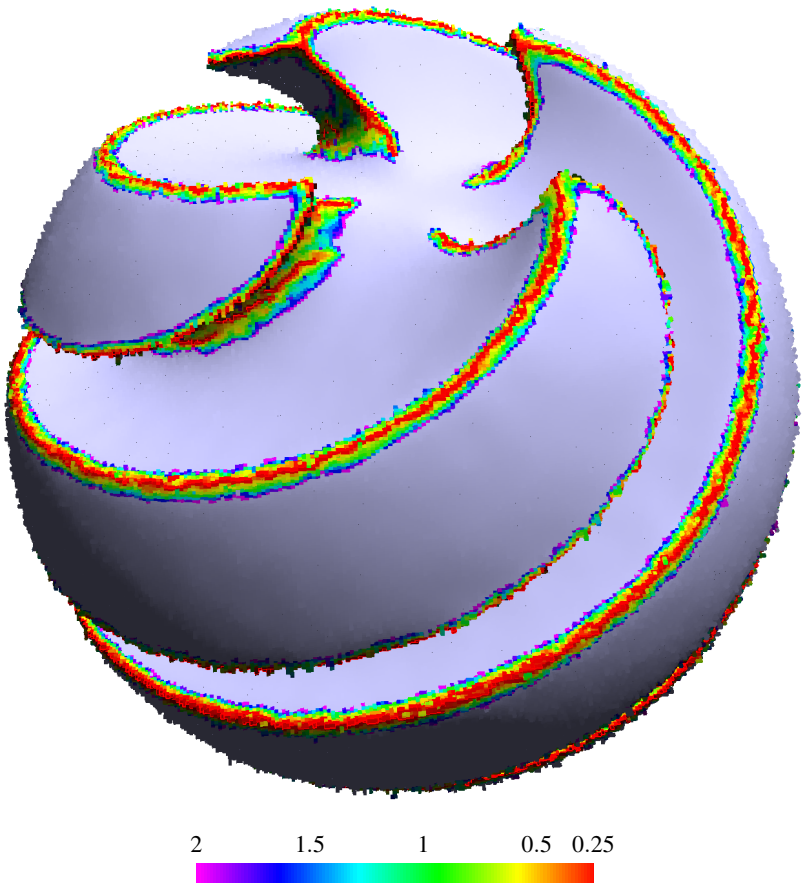
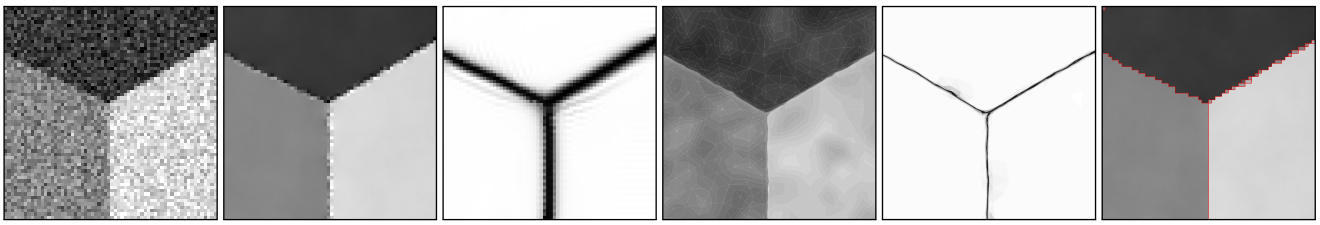
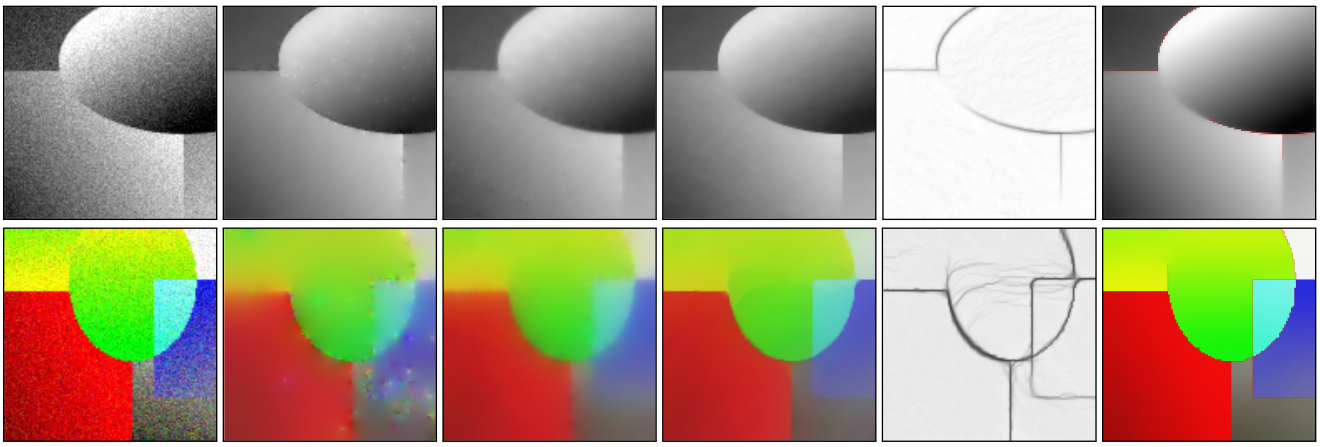


Figure 8: *Illustration of the progressive localization of sharp features on SharpSphere* 256^3 *with noise* $k = 0.4$. *Input normal field* $\mathbf{g}$ *is estimated with II,* $r = 8$. *AT functional is computed with* $\alpha = 0.05$, $\lambda = 0.01$, *for a range* $\varepsilon \in [2, 0.25]$. *Each quad element is colored according to the smallest value of* ε *for which* $\mathbf{v}$ *was below* $\frac{1}{2}$ *on at least two of its vertices. If a quad was never classified as "feature" then it displayed in bluish gray.*



g (a) FD, *u* (b) FD, *v* (c) FE, *u* (d) FE, *v* (e) our approach

Figure 9: *Minimization of AT with finite differences (FD), finite elements (FE) from [BC00] and our approach on a triple point image with PSNR*= 20.3*dB*.



g (a) McCormick relax. *u* [KK14] (b) Quadratic relax. *u* [KK14] (c) Linear relax. *u* [KK14] (d) *v* [KK14] (e) our approach

Figure 10: *Comparison of the minimization of convex relaxations of AT (extracted from [KK14]) and our approach on images extracted from [KK14].*