Discrete calculus model of Ambrosio-Tortorelli's functional

Jacques-Olivier Lachaud[1]

[1]Lab. of Mathematics, University Savoie Mont Blanc

May 23, 2019

Autour des mathématiques, 80 ans du CNRS

Laboratoire de Mathématiques

Collaborators


Marion Foare

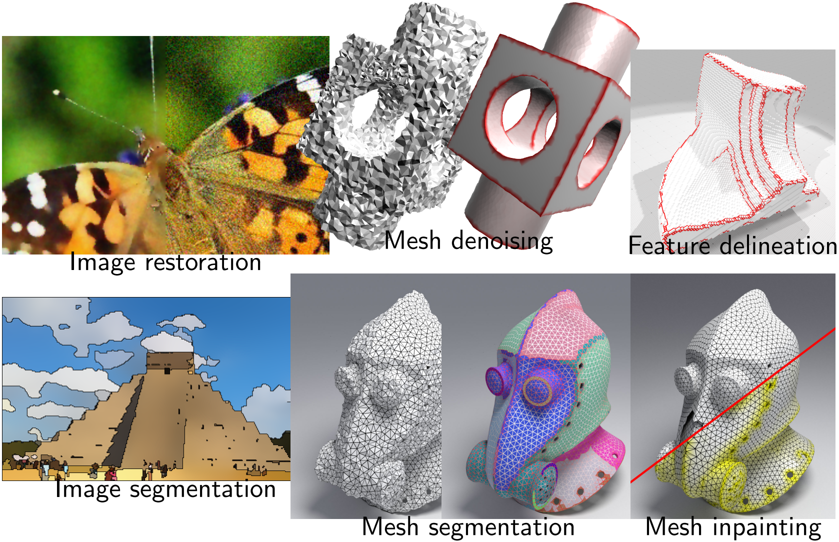

David Coeurjolly


Pierre Gueth


Hugues Talbot

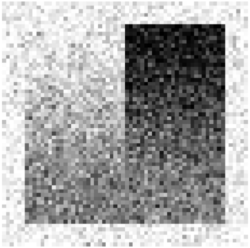

Nicolas Bonneel

A versatile tool for piecewise smooth image and geometry processing

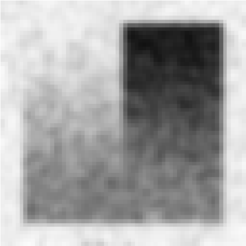


Discrete calculus model of Ambrosio-Tortorelli's functionnal

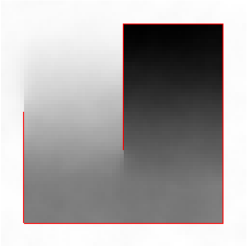
Piecewise smooth reconstruction of functions/images



Noisy input g

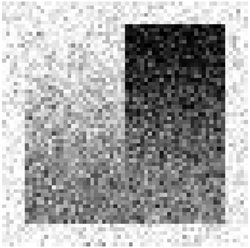


smooth reco.

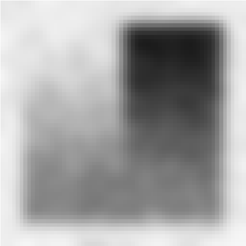


piecewise smooth reco.

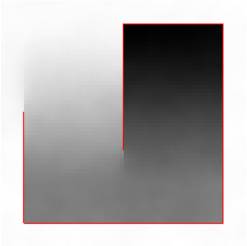
Piecewise smooth reconstruction of functions/images



Noisy input g

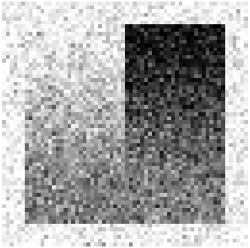


smooth reco.

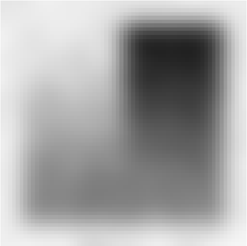


piecewise smooth reco.

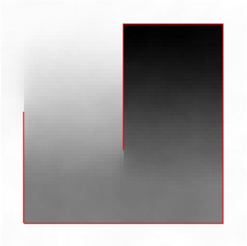
Piecewise smooth reconstruction of functions/images



Noisy input g



smooth reco.



piecewise smooth reco.

Mumford-Shah functional

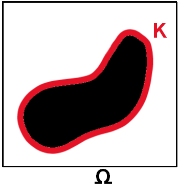
[Mumford and Shah, 1989]

Mumford-Shah functional for image restoration

We minimize

$$\mathcal{MS}(K,u) = \alpha \underbrace{\int_{\Omega\setminus K} |u-g|^2 \, \mathrm{dx}}_{\text{fidelity term}} + \underbrace{\int_{\Omega\setminus K} |\nabla u|^2 \, \mathrm{dx}}_{\text{smoothness term}} + \lambda \underbrace{\mathcal{H}^1(K\cap\Omega)}_{\text{discontinuities length}}$$

- Ω the image domain
- g the input image
- u a piecewise smooth approximation of g
- K the set of discontinuities
- $\mathcal{H}^1$ the Hausdorff measure



Mumford-Shah functional

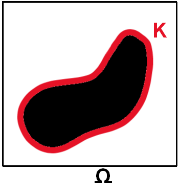
[Mumford and Shah, 1989]

Mumford-Shah functional for image restoration

We minimize

$$\mathcal{MS}(K,u) = \alpha \underbrace{\int_{\Omega\setminus K} |u-g|^2 \, \mathrm{d}x}_{\text{fidelity term}} + \underbrace{\int_{\Omega\setminus K} |\nabla u|^2 \, \mathrm{d}x}_{\text{smoothness term}} + \lambda \underbrace{\mathcal{H}^1(K\cap\Omega)}_{\text{discontinuities length}}$$

- Ω the image domain
- g the input image
- u a piecewise smooth approximation of g
- ✗ K the set of discontinuities
- ✗ $\mathcal{H}^1$ the Hausdorff measure



Mumford-Shah functional

[Mumford and Shah, 1989]

Notably difficult to minimize

Many relaxations and convexifications have been proposed.

- Total Variation [Rudin et al., 1992] and its variants
- Multi-phase level sets [Vese and Chan, 2002] and follow-ups
- Discrete graph approaches [Boykov et al., 2001, Boykov and Funka-Lea, 2006]
- Calibration method [Alberti et al., 2003] and associated algorithms [Pock et al., 2009, Chambolle and Pock, 2011]
- Ambrosio-Tortorelli functional [Ambrosio and Tortorelli, 1992]
- convex relaxations of AT [Kee and Kim, 2014]

Mumford-Shah functional

[Mumford and Shah, 1989]

Notably difficult to minimize

Many relaxations and convexifications have been proposed.

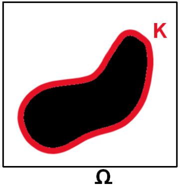
- Total Variation [Rudin et al., 1992] and its variants
- Multi-phase level sets [Vese and Chan, 2002] and follow-ups
- Discrete graph approaches [Boykov et al., 2001, Boykov and Funka-Lea, 2006]
- Calibration method [Alberti et al., 2003] and associated algorithms [Pock et al., 2009, Chambolle and Pock, 2011]
- Ambrosio-Tortorelli functional [Ambrosio and Tortorelli, 1992]
- convex relaxations of AT [Kee and Kim, 2014]

Ambrosio-Tortorelli functional

[Ambrosio and Tortorelli, 1992]

$$AT_\varepsilon(u, v) = \alpha \int_\Omega |u-g|^2 \,\mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \,\mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \frac{1}{\varepsilon} \frac{(1-v)^2}{4} \,\mathrm{dx}$$

- Ω the image domain
- g the input image
- u a piecewise smooth approximation of g
- v a smooth approximation of $1 - \chi_K$
- ✓ whole domain integration
- ✓ no Hausdorff measure



Ambrosio-Tortorelli functional

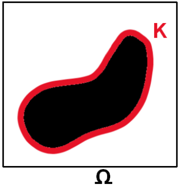
[Ambrosio and Tortorelli, 1992]

$$AT_\varepsilon(u, v) = \alpha \int_\Omega |u - g|^2 \, \mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \, \mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \frac{1}{\varepsilon} \frac{(1 - v)^2}{4} \, \mathrm{dx}$$

- Ω the image domain
- g the input image
- u a piecewise smooth approximation of g
- v a smooth approximation of $1 - \chi_K$

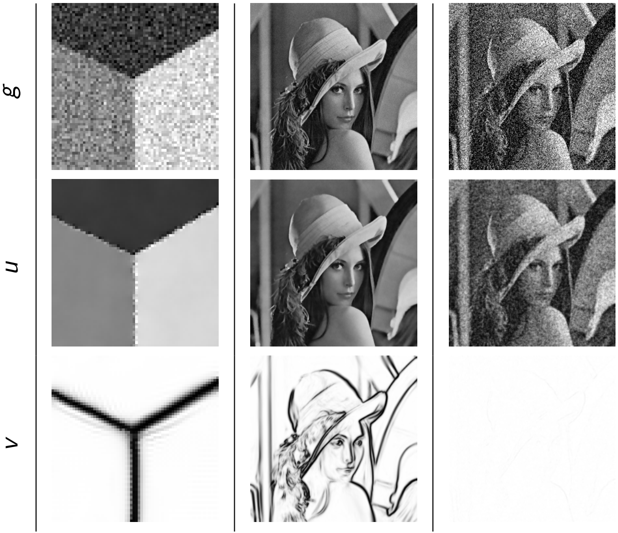
✓ whole domain integration

✓ no Hausdorff measure

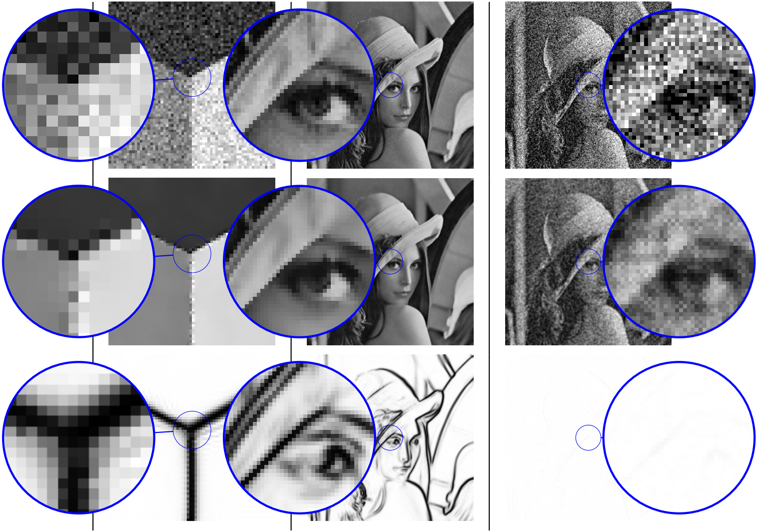


$$\Gamma\text{-convergence:} \quad AT_\varepsilon \xrightarrow[\varepsilon \to 0]{\Gamma} \mathcal{MS}$$

Finite differences implementation

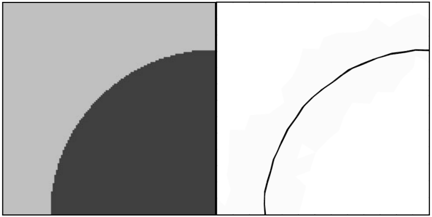


Finite differences implementation



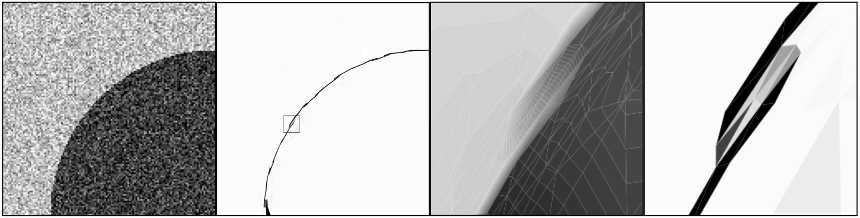
Finite elements implementation

- Proposed in [Bourdin and Chambolle, 2000]



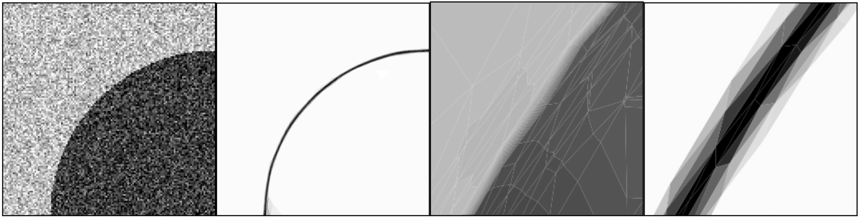
Finite elements implementation

- Proposed in [Bourdin and Chambolle, 2000]



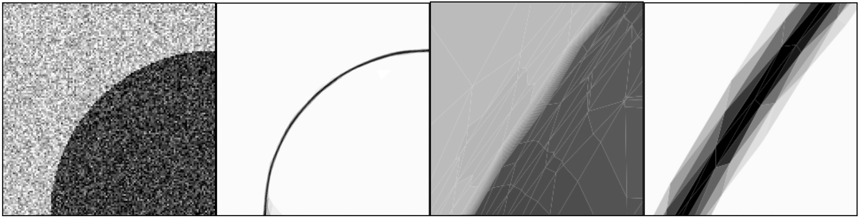
Finite elements implementation

- ▷ Proposed in [Bourdin and Chambolle, 2000]
- ▷ Finite elements with mesh refinement and realignment



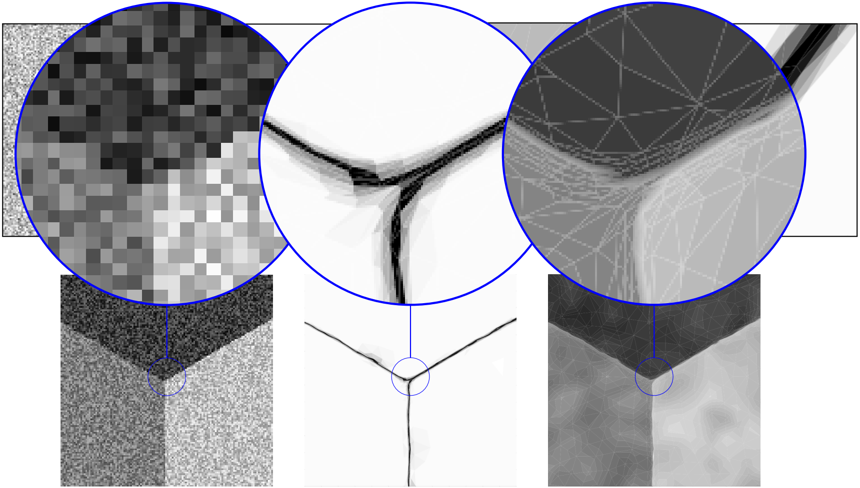
Finite elements implementation

- Proposed in [Bourdin and Chambolle, 2000]
- Finite elements with mesh refinement and realignment



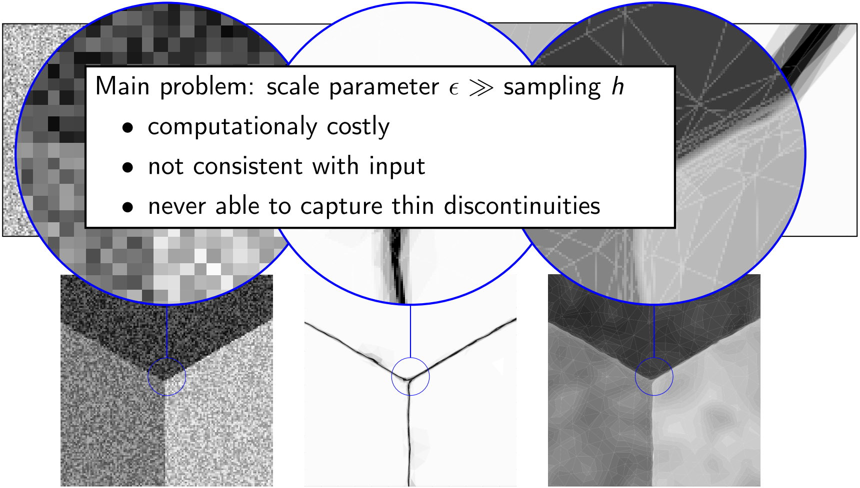
Finite elements implementation

- Proposed in [Bourdin and Chambolle, 2000]
- Finite elements with mesh refinement and realignment



Finite elements implementation

- Proposed in [Bourdin and Chambolle, 2000]
- Finite elements with mesh refinement and realignment

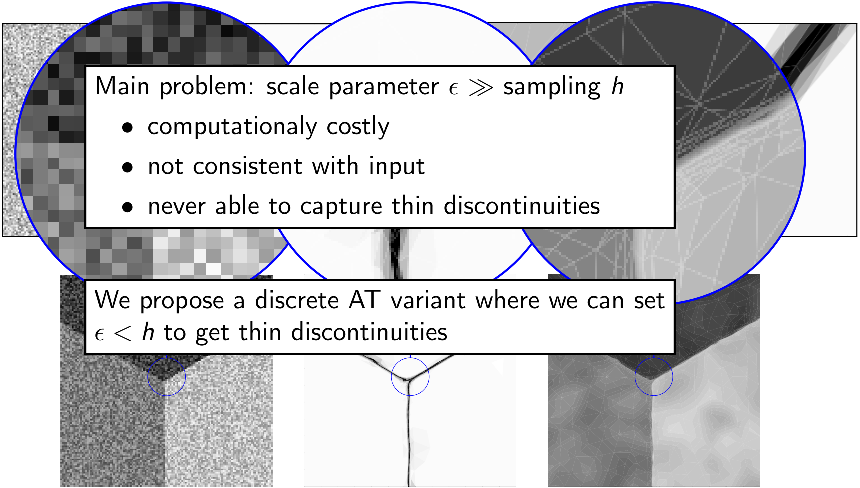


Main problem: scale parameter $\epsilon \gg$ sampling h

- computationaly costly
- not consistent with input
- never able to capture thin discontinuities

Finite elements implementation

- Proposed in [Bourdin and Chambolle, 2000]
- Finite elements with mesh refinement and realignment



Main problem: scale parameter $\epsilon \gg$ sampling h

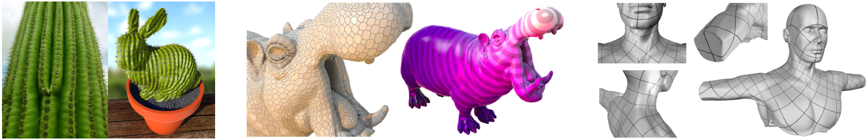
- computationaly costly
- not consistent with input
- never able to capture thin discontinuities

We propose a discrete AT variant where we can set $\epsilon < h$ to get thin discontinuities

Discrete calculus model of Ambrosio-Tortorelli's functionnal

Discrete Calculus

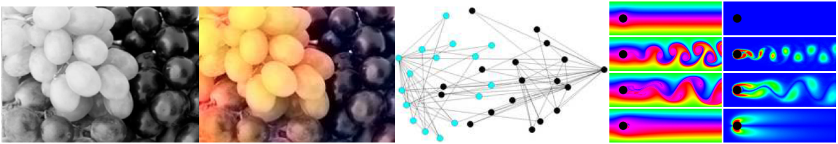
Computer graphics, geometry processing, shape optimization



(Images: Knöppel et al. 2015, Crane et al. 2013, Springborn et al. 2010)

Discrete exterior calculus [Desbrun, Hirani, Leok, . . .]
Discrete differential calculus [Polthier, Pinkall, Bobenko, . . .]
Discrete calculus [Grady, Polimeni, . . .]

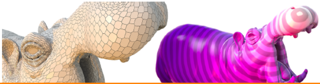
Graph and network analysis, image processing, fluid simul.

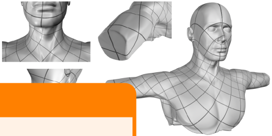


(Images: Bugeau et al. 2014, couprie et al. 2014, Elcott et al. 2006)

Discrete Calculus

Computer graphics, geometry processing, shape optimization

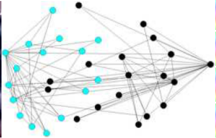


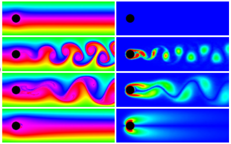


Discrete exterior calculus (DEC)

- no discretization, discrete by nature
- keep algebraic properties of calculus, exact Stokes' theorem
- reduces to matrix/vectors
- works without embedding, just metric
- "any" cell complex, arbitrary dimension

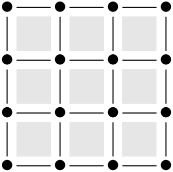
Graph and network analysis, image processing, fluid simul.





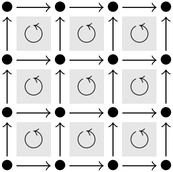
(Images: Bugeau et al. 2014, couprie et al. 2014, Elcott et al. 2006)

Cell complex, chains, boundary, forms



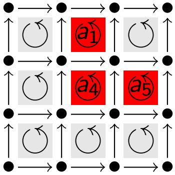
- *cell complex K*: vertices, edges, faces (pixels)

Cell complex, chains, boundary, forms



- *cell complex* K: vertices, edges, faces (pixels) with orientation

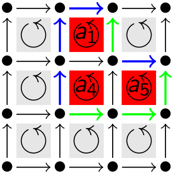
Cell complex, chains, boundary, forms



$\sigma := a_1 + a_4 + a_5 \in C_2(K)$

- *cell complex* K: vertices, edges, faces (pixels) with orientation
- *k-chains*: $C_k(K)$ are integral formal sums of oriented cells

Cell complex, chains, boundary, forms



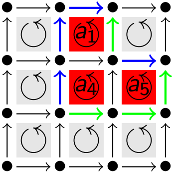
$$\sigma := a_1 + a_4 + a_5 \in C_2(K)$$

$$\eta := \sum e_i - \sum e_j \in C_1(K)$$

$$\eta = \partial_2 \sigma$$

- *cell complex* K: vertices, edges, faces (pixels) with orientation
- *k-chains*: $C_k(K)$ are integral formal sums of oriented cells
- *boundary operators*: $\cdots C_2(K) \xrightarrow{\partial_2} C_1(K) \xrightarrow{\partial_1} C_0(K) \xrightarrow{\partial_0} 0$

Cell complex, chains, boundary, forms



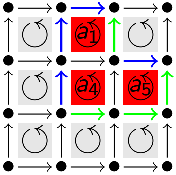
$\sigma := a_1 + a_4 + a_5 \in C_2(K)$

$\eta := \sum e_i - \sum e_j \in C_1(K)$

$\eta = \partial_2 \sigma$

- *cell complex* K: vertices, edges, faces (pixels) with orientation
- *k-chains*: $C_k(K)$ are integral formal sums of oriented cells
- *boundary operators*: $\cdots C_2(K) \xrightarrow{\partial_2} C_1(K) \xrightarrow{\partial_1} C_0(K) \xrightarrow{\partial_0} 0$
- *discrete k-forms*: elements of $C^k(K) := \mathrm{Hom}(C_k(K), \mathbb{R})$
  - 0-forms: functions, i.e. a value per vertex
  - 1-forms: differential forms/vector field, i.e. a value per edge
  - 2-forms: area forms, i.e. a value per face

Cell complex, chains, boundary, forms



$\sigma := a_1 + a_4 + a_5 \in C_2(K)$

$\eta := \sum e_i - \sum e_j \in C_1(K)$

$\eta = \partial_2 \sigma$

- *cell complex* K: vertices, edges, faces (pixels) with orientation
- *k-chains*: $C_k(K)$ are integral formal sums of oriented cells
- *boundary operators*: $\cdots C_2(K) \xrightarrow{\partial_2} C_1(K) \xrightarrow{\partial_1} C_0(K) \xrightarrow{\partial_0} 0$
- *discrete k-forms*: elements of $C^k(K) := \mathrm{Hom}(C_k(K), \mathbb{R})$
  - 0-forms: functions, i.e. a value per vertex
  - 1-forms: differential forms/vector field, i.e. a value per edge
  - 2-forms: area forms, i.e. a value per face
- *Integral* $\int_\sigma \alpha$ = pairing k-form α with k-chain σ

$$\int_\sigma \alpha := \alpha(\sigma) = \sum_i a_i \alpha(c_i) \quad \text{if } \sigma = \sum_i a_i c_i$$

Exterior derivative, Stokes theorem

- *exterior derivative* defined by duality: $\mathbf{d}_k : C^k(K) \to C^{k+1}(K)$

$$(\mathbf{d}_k \alpha^k)(\sigma_{k+1}) := \alpha^k(\partial_{k+1}\sigma_{k+1})$$

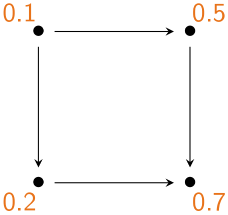
  thus incidence relations define derivative by duality

Exterior derivative, Stokes theorem

- *exterior derivative* defined by duality: $\mathbf{d}_k : C^k(K) \to C^{k+1}(K)$

$$(\mathbf{d}_k \alpha^k)(\sigma_{k+1}) := \alpha^k(\partial_{k+1}\sigma_{k+1})$$

  thus incidence relations define derivative by duality



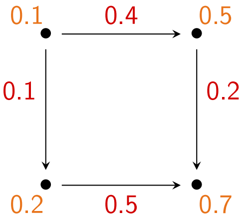
- Function or discrete 0-form : $\alpha = (0.2, 0.7, 0.1, 0.5)$

Exterior derivative, Stokes theorem

- *exterior derivative* defined by duality: $\mathbf{d}_k : C^k(K) \to C^{k+1}(K)$

$$(\mathbf{d}_k \alpha^k)(\sigma_{k+1}) := \alpha^k(\partial_{k+1}\sigma_{k+1})$$

thus incidence relations define derivative by duality



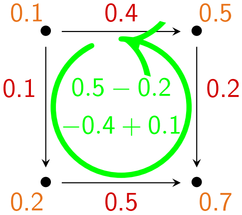
- Function or discrete 0-form : $\alpha = (0.2, 0.7, 0.1, 0.5)$
- 1-form $\mathbf{d}_0(\alpha) = \beta = (0.5, 0.1, 0.2, 0.4)$

Exterior derivative, Stokes theorem

- *exterior derivative* defined by duality: $\mathbf{d}_k : C^k(K) \to C^{k+1}(K)$

$$(\mathbf{d}_k \alpha^k)(\sigma_{k+1}) := \alpha^k(\partial_{k+1}\sigma_{k+1})$$

  thus incidence relations define derivative by duality



- Function or discrete 0-form : $\alpha = (0.2, 0.7, 0.1, 0.5)$
- 1-form $\mathbf{d}_0(\alpha) = \beta = (0.5, 0.1, 0.2, 0.4)$
- 2-form $\mathbf{d}_1(\beta) = 0$, since $\mathbf{d}_1\mathbf{d}_0 = 0$

Exterior derivative, Stokes theorem

- *exterior derivative* defined by duality: $\mathbf{d}_k : C^k(K) \to C^{k+1}(K)$

$$(\mathbf{d}_k \alpha^k)(\sigma_{k+1}) := \alpha^k(\partial_{k+1}\sigma_{k+1})$$

thu

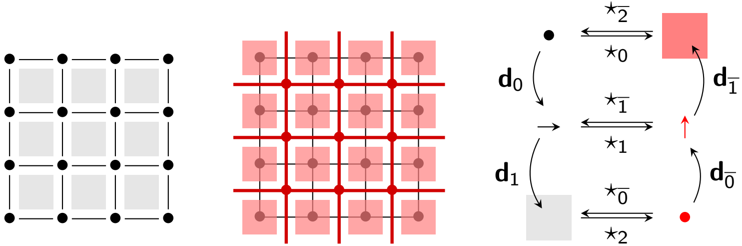
(discrete) Stokes theorem is trivial by definition

$$\int_\sigma \mathbf{d}\alpha = \int_{\partial\sigma} \alpha$$

for σ any k-chain and α any $k-1$-form

- Function or discrete 0-form : $\alpha = (0.2, 0.7, 0.1, 0.5)$
- 1-form $\mathbf{d}_0(\alpha) = \beta = (0.5, 0.1, 0.2, 0.4)$
- 2-form $\mathbf{d}_1(\beta) = 0$, since $\mathbf{d}_1\mathbf{d}_0 = 0$

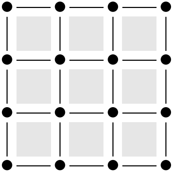
Dual cell complex, Hodge star, calculus



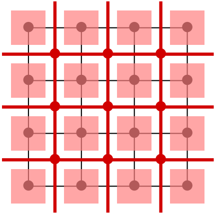
complex K dual complex $\overline{K}$ primal   dual

- Hodge duality created with dual/orthogonal structure

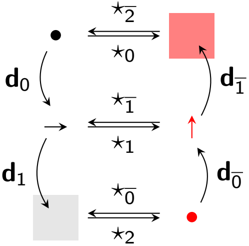
Dual cell complex, Hodge star, calculus



complex K



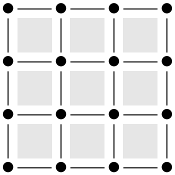
dual complex $\overline{K}$



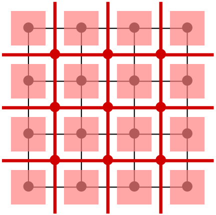
primal dual

- Hodge duality created with dual/orthogonal structure
- anti-derivatives $\mathbf{d}_{\overline{k}}$ are derivatives in dual complex
  - ▷ in matrix form $\mathbf{d}_{\overline{k}}^{\mathsf{T}} := \mathbf{d}_{n-1-k}$

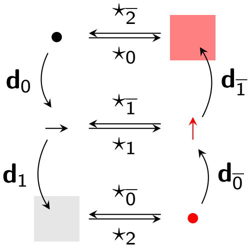
Dual cell complex, Hodge star, calculus



complex K



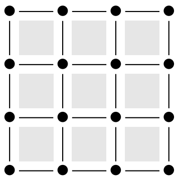
dual complex $\overline{K}$



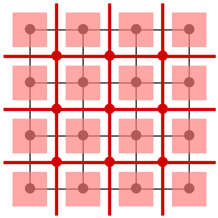
primal dual

- Hodge duality created with dual/orthogonal structure
- anti-derivatives $\mathbf{d}_{\overline{k}}$ are derivatives in dual complex
  - ▷ in matrix form $\mathbf{d}_{\overline{k}}^{\mathsf{T}} := \mathbf{d}_{n-1-k}$
- Hodge stars $\star_k$ transport k-forms to dual $2-k$-forms
  - ▷ diagonal matrices incorporating metric information
  - ▷ e.g. $\star_k \mathbf{1} = \alpha$ is the area 2-form $\mathrm{d}A$

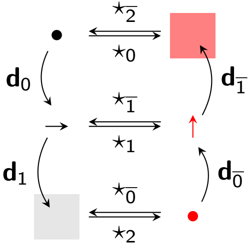
Dual cell complex, Hodge star, calculus



complex K



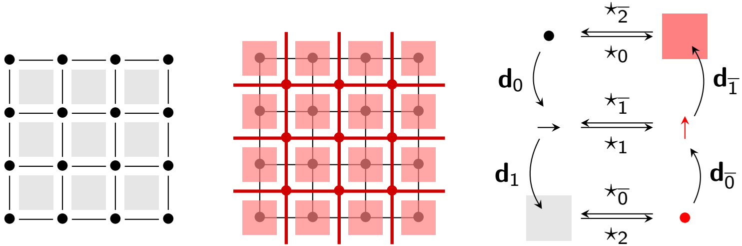
dual complex $\overline{K}$



primal dual

- Hodge duality created with dual/orthogonal structure
- anti-derivatives $\mathbf{d}_{\overline{k}}$ are derivatives in dual complex
  - ▷ in matrix form $\mathbf{d}_{\overline{k}}^{\mathsf{T}} := \mathbf{d}_{n-1-k}$
- Hodge stars $\star_k$ transport k-forms to dual $2-k$-forms
  - ▷ diagonal matrices incorporating metric information
  - ▷ e.g. $\star_k \mathbf{1} = \alpha$ is the area 2-form $\mathrm{d}A$
- wedge products satisfy algebraic properties (Leibniz rules . . .)
  - ▷ $\alpha \wedge \beta := \mathrm{diag}(\alpha)\beta$, for $\alpha \in C^k(K), \beta \in C^{2-k}(\overline{K})$,
  - ▷ $f \wedge \gamma := \mathrm{diag}(\mathbf{M}_{01} f)\gamma$, for $f \in C^0(K), \gamma \in C^1(K)$ . . .

Dual cell complex, Hodge star, calculus



Almost all the calculus is built from the previous operators

- codifferentials $\delta_1 := -\star_{\bar{2}}\mathbf{d}_{\bar{1}}\star_1$, $\delta_2 := -\star_{\bar{1}}\mathbf{d}_{\bar{0}}\star_2$,
- Laplacian $\Delta := \delta_1 \mathbf{d}_0$
- Edge Laplacian $\Delta_1 := \mathbf{d}_0\delta_1 + \delta_2\mathbf{d}_1$,
- musical ops : Vector field $\xrightarrow{\flat}$ 1-form $\xrightarrow{\sharp}$ Vector field
- gradient $\nabla f := (\mathbf{d}_0 f)^\sharp$
- divergence $\mathrm{div}\mathbf{V} := \delta_1 \mathbf{V}^\flat$
- L^2 inner-product $(\alpha, \beta)_{\Omega,k} := \int_\Omega \alpha \wedge \star_k \beta$, for α, β k-forms

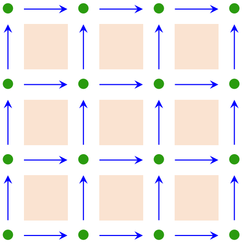
▷ $f \wedge \gamma := \mathrm{diag}(\mathbf{M}_{01} f)\gamma$, for $f \in C^0(K), \gamma \in C^1(K)$ ...

Discrete calculus model of Ambrosio-Tortorelli's functionnal

Discrete formulation of AT

On faces and vertices

$$AT_\varepsilon(u, v) = \alpha \int_\Omega |u - g|^2 \ \mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \ \mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \frac{1}{4\varepsilon}(1 - v)^2 \ \mathrm{dx}$$



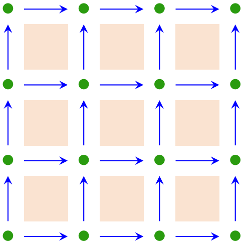
We choose :

- functions u, g to live on faces
  - ▷ u, g are 2-forms
  - ▷ equivalently dual 0-forms
- function v to live on vertices
  - ▷ v is a 0-form

Discrete formulation of AT

On faces and vertices

$$AT_\varepsilon(u,v) = \alpha \int_\Omega |u-g|^2 \,\mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \,\mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \frac{1}{4\varepsilon}(1-v)^2 \,\mathrm{dx}$$



We choose :

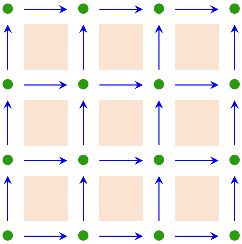
- functions u, g to live on faces
  - ▷ u, g are 2-forms
  - ▷ equivalently dual 0-forms
- function v to live on vertices
  - ▷ v is a 0-form

$$\mathrm{AT}_\varepsilon^{2,0}(\mathrm{u},\mathrm{v}) = \alpha\,(\mathrm{u}-\mathrm{g},\mathrm{u}-\mathrm{g})_{\Omega,2}$$

Discrete formulation of AT

On faces and vertices

$$AT_\varepsilon(u,v) = \alpha \int_\Omega |u-g|^2 \, \mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \, \mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \frac{1}{4\varepsilon}(1-v)^2 \, \mathrm{dx}$$



We choose :

- functions u, g to live on faces
  - ▷ u, g are 2-forms
  - ▷ equivalently dual 0-forms
- function v to live on vertices
  - ▷ v is a 0-form

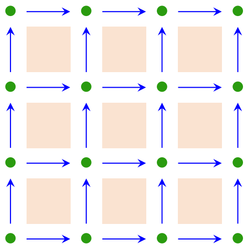
$$\mathrm{AT}_\varepsilon^{2,0}(\mathrm{u},\mathrm{v}) = \alpha(\mathrm{u}-\mathrm{g},\mathrm{u}-\mathrm{g})_{\Omega,2}$$

$$+ \; \lambda\varepsilon\,(\mathbf{d}_0\mathrm{v},\mathbf{d}_0\mathrm{v})_{\Omega,1}$$

Discrete formulation of AT

On faces and vertices

$$AT_\varepsilon(u,v) = \alpha \int_\Omega |u-g|^2 \ \mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \ \mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \tfrac{1}{4\varepsilon}(1-v)^2 \ \mathrm{dx}$$



We choose :

- functions u, g to live on faces
  - ▷ u, g are 2-forms
  - ▷ equivalently dual 0-forms
- function v to live on vertices
  - ▷ v is a 0-form

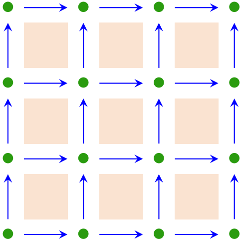
$$\mathrm{AT}_\varepsilon^{2,0}(\mathrm{u},\mathrm{v}) = \alpha\,(\mathrm{u}-\mathrm{g},\mathrm{u}-\mathrm{g})_{\Omega,2}$$

$$+\ \lambda\varepsilon\,(\mathbf{d}_0\mathrm{v},\mathbf{d}_0\mathrm{v})_{\Omega,1}\ +\ \frac{\lambda}{4\varepsilon}\,(1-\mathrm{v},1-\mathrm{v})_{\Omega,0}$$

Discrete formulation of AT

On faces and vertices

$$AT_\varepsilon(u,v) = \alpha \int_\Omega |u-g|^2 \, \mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \, \mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \frac{1}{4\varepsilon}(1-v)^2 \, \mathrm{dx}$$



We choose :

- functions u, g to live on faces
  - ▷ u, g are 2-forms
  - ▷ equivalently dual 0-forms
- function v to live on vertices
  - ▷ v is a 0-form

$$\mathrm{AT}_\varepsilon^{2,0}(\mathrm{u},\mathrm{v}) = \alpha\,(\mathrm{u}-\mathrm{g},\mathrm{u}-\mathrm{g})_{\Omega,2} + (?,?)_\Omega$$

$$+ \lambda\varepsilon\,(\mathbf{d}_0\mathrm{v},\mathbf{d}_0\mathrm{v})_{\Omega,1} + \frac{\lambda}{4\varepsilon}\,(1-\mathrm{v},1-\mathrm{v})_{\Omega,0}$$

Discrete formulation of AT

Cross term mixing u and v

$$\int_\Omega v^2 |\nabla u|^2 \;\mathrm{dx} \quad = (\mathsf{v}\delta_2\mathsf{u}, \mathsf{v}\delta_2\mathsf{u})_{\Omega,1}$$

- $\mathsf{v}\delta_2\mathsf{u} = \mathsf{v} \wedge \delta_2\mathsf{u} = \mathrm{diag}(\mathbf{M}_{01}\mathsf{v})\delta_2\mathsf{u}$

Discrete formulation of AT

Cross term mixing u and v

$$\int_\Omega v^2 |\nabla u|^2 \ \mathrm{dx} \quad = (\mathsf{v}\delta_2\mathsf{u}, \mathsf{v}\delta_2\mathsf{u})_{\Omega,1}$$

- $\mathsf{v}\delta_2\mathsf{u} = \mathsf{v} \wedge \delta_2\mathsf{u} = \mathrm{diag}(\mathbf{M}_{01}\mathsf{v})\delta_2\mathsf{u}$

0.8	0.8	1.0
1.0	0.0	0.2
1.0	0.2	0.8

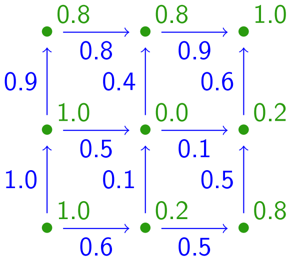
- 0-form v

Discrete formulation of AT

Cross term mixing u and v

$$\int_\Omega v^2 |\nabla u|^2 \;\mathrm{dx} \quad = (\mathrm{v}\delta_2 \mathrm{u}, \mathrm{v}\delta_2 \mathrm{u})_{\Omega,1}$$

- $\mathrm{v}\delta_2\mathrm{u} = \mathrm{v} \wedge \delta_2 \mathrm{u} = \mathrm{diag}(\mathbf{M}_{01}\mathrm{v})\delta_2\mathrm{u}$



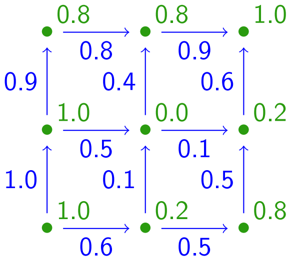
- 0-form v
- 1-form $\mathbf{M}_{01}\mathrm{v}$

Discrete formulation of AT

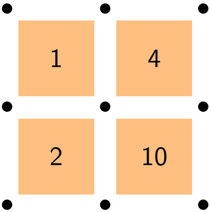
Cross term mixing u and v

$$\int_\Omega v^2 |\nabla u|^2 \; \mathrm{dx} \quad = (\mathsf{v}\delta_2 \mathsf{u}, \mathsf{v}\delta_2 \mathsf{u})_{\Omega,1}$$

- $\mathsf{v}\delta_2\mathsf{u} = \mathsf{v} \wedge \delta_2 \mathsf{u} = \mathrm{diag}(\mathbf{M}_{01}\mathsf{v})\delta_2\mathsf{u}$



- 0-form v
- 1-form $\mathbf{M}_{01}\mathsf{v}$



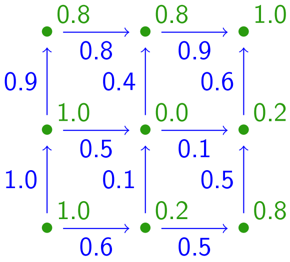
- 2-form u

Discrete formulation of AT

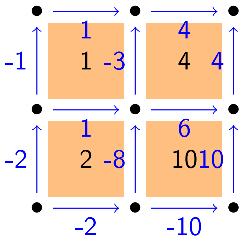
Cross term mixing u and v

$$\int_\Omega v^2 |\nabla u|^2 \ \mathrm{dx} \quad = (\mathsf{v}\delta_2\mathsf{u}, \mathsf{v}\delta_2\mathsf{u})_{\Omega,1}$$

- $\mathsf{v}\delta_2\mathsf{u} = \mathsf{v} \wedge \delta_2\mathsf{u} = \mathrm{diag}(\mathbf{M}_{01}\mathsf{v})\delta_2\mathsf{u}$



- 0-form v
- 1-form $\mathbf{M}_{01}\mathsf{v}$



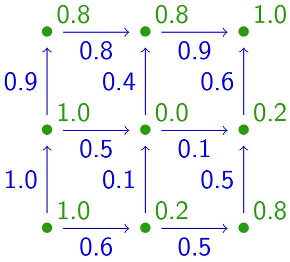
- 2-form u
- 1-form $\delta_2\mathsf{u}$

Discrete formulation of AT

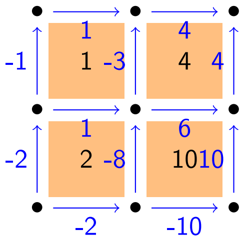
Cross term mixing u and v

$$\int_\Omega v^2 |\nabla u|^2 \ \mathrm{dx} \quad = (\mathsf{v}\delta_2\mathsf{u}, \mathsf{v}\delta_2\mathsf{u})_{\Omega,1}$$

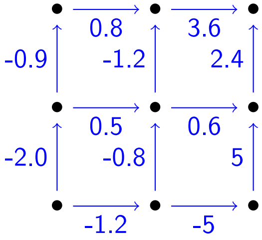
- $\mathsf{v}\delta_2\mathsf{u} = \mathsf{v} \wedge \delta_2\mathsf{u} = \mathrm{diag}(\mathbf{M}_{01}\mathsf{v})\delta_2\mathsf{u}$



- 0-form v
- 1-form $\mathbf{M}_{01}\mathsf{v}$



- 2-form u
- 1-form $\delta_2\mathsf{u}$



- 1-form $\mathrm{diag}(\mathbf{M}_{01}\mathsf{v})\delta_2\mathsf{u}$

Discrete formulation of AT

$$\mathrm{AT}_{\varepsilon}^{2,0}(\mathsf{u},\mathsf{v}) = \alpha\,(\mathsf{u}-\mathsf{g},\mathsf{u}-\mathsf{g})_{\Omega,2} + (\mathsf{v}\wedge\delta_2\mathsf{u},\mathsf{v}\wedge\delta_2\mathsf{u})_{\Omega,1}$$
$$+\,\lambda\varepsilon\,(\mathbf{d}_0\mathsf{v},\mathbf{d}_0\mathsf{v})_{\Omega,1} + \frac{\lambda}{4\varepsilon}\,(1-\mathsf{v},1-\mathsf{v})_{\Omega,0}$$

Discrete formulation of AT

$$\mathrm{AT}_\varepsilon^{2,0}(u, v) = \alpha\,(u - g, u - g)_{\Omega,2} + (v \wedge \delta_2 u, v \wedge \delta_2 u)_{\Omega,1}$$
$$+ \lambda\varepsilon\,(\mathbf{d}_0 v, \mathbf{d}_0 v)_{\Omega,1} + \frac{\lambda}{4\varepsilon}\,(1 - v, 1 - v)_{\Omega,0}$$

- with matrices $\mathbf{A} := \mathbf{d}_0$, $\mathbf{B}' := \delta_2$, $\mathbf{G}_k := \star_k$.

$$\mathrm{AT}_\varepsilon^{2,0}(u, v) = \alpha(u - g)^\mathsf{T}\mathbf{G}_2(u - g) + u^\mathsf{T}\mathbf{B}'^\mathsf{T}\mathrm{diag}(\mathbf{M}_{01}v)^2\mathbf{G}_1\mathbf{B}'u$$
$$+ \lambda\varepsilon v^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{G}_1\mathbf{A}v + \frac{\lambda}{4\varepsilon}(1 - v)^\mathsf{T}\mathbf{G}_0(1 - v)$$

Discrete formulation of AT

$$\mathrm{AT}_{\varepsilon}^{2,0}(u,v) = \alpha\,(u-g,u-g)_{\Omega,2} + (v\wedge\delta_2 u, v\wedge\delta_2 u)_{\Omega,1} + \lambda\varepsilon\,(\mathbf{d}_0 v,\mathbf{d}_0 v)_{\Omega,1} + \frac{\lambda}{4\varepsilon}\,(1-v,1-v)_{\Omega,0}$$

- with matrices $\mathbf{A} := \mathbf{d}_0$, $\mathbf{B}' := \delta_2$, $\mathbf{G}_k := \star_k$.

$$\mathrm{AT}_{\varepsilon}^{2,0}(u,v) = \alpha(u-g)^{\mathsf{T}}\mathbf{G}_2(u-g) + u^{\mathsf{T}}\mathbf{B}'^{\mathsf{T}}\mathrm{diag}(\mathbf{M}_{01}v)^2\mathbf{G}_1\mathbf{B}'u + \lambda\varepsilon v^{\mathsf{T}}\mathbf{A}^{\mathsf{T}}\mathbf{G}_1\mathbf{A}v + \frac{\lambda}{4\varepsilon}(1-v)^{\mathsf{T}}\mathbf{G}_0(1-v)$$

- Euler-Lagrange: $\min_{u,v}\mathrm{AT}_{\varepsilon}^{2,0} \Rightarrow \frac{\mathrm{dAT}_{\varepsilon}^{2,0}}{\mathrm{d}u} = 0$ and $\frac{\mathrm{dAT}_{\varepsilon}^{2,0}}{\mathrm{d}v} = 0$
- $\mathrm{AT}_{\varepsilon}^{2,0}$ is quadratic in u and in v

Discrete formulation of AT

$$\mathrm{AT}_\varepsilon^{2,0}(u,v) = \alpha\,(u-g,u-g)_{\Omega,2} + (v\wedge\delta_2 u, v\wedge\delta_2 u)_{\Omega,1} + \lambda\varepsilon\,(\mathbf{d}_0 v,\mathbf{d}_0 v)_{\Omega,1} + \frac{\lambda}{4\varepsilon}\,(1-v,1-v)_{\Omega,0}$$

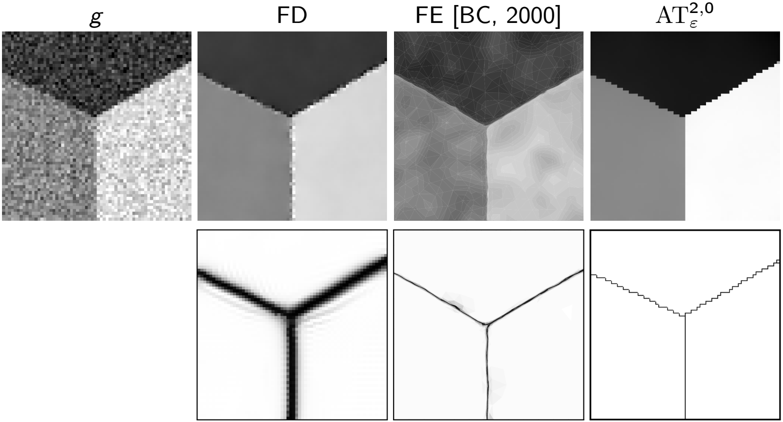
- with matrices $\mathbf{A} := \mathbf{d}_0$, $\mathbf{B}' := \delta_2$, $\mathbf{G}_k := \star_k$.

$$\mathrm{AT}_\varepsilon^{2,0}(u,v) = \alpha(u-g)^\intercal\mathbf{G}_2(u-g) + u^\intercal\mathbf{B}'^\intercal\mathrm{diag}(\mathbf{M}_{01}v)^2\mathbf{G}_1\mathbf{B}'u + \lambda\varepsilon v^\intercal\mathbf{A}^\intercal\mathbf{G}_1\mathbf{A}v + \frac{\lambda}{4\varepsilon}(1-v)^\intercal\mathbf{G}_0(1-v)$$

- Euler-Lagrange: $\min_{u,v}\mathrm{AT}_\varepsilon^{2,0} \Rightarrow \frac{\mathrm{dAT}_\varepsilon^{2,0}}{\mathrm{d}u} = 0$ and $\frac{\mathrm{dAT}_\varepsilon^{2,0}}{\mathrm{d}v} = 0$
- $\mathrm{AT}_\varepsilon^{2,0}$ is quadratic in u and in v
- We solve alternatively for u and v the sparse linear systems:

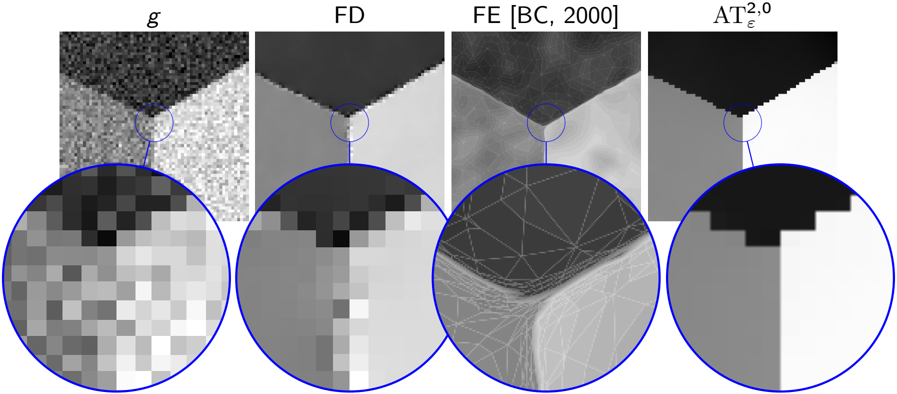
$$\begin{cases} \left[\alpha\mathbf{G}_2 - \mathbf{B}'^\intercal\mathrm{diag}\left(\mathbf{M}_{01}v\right)^2\mathbf{G}_1\mathbf{B}'\right]u = \alpha\mathbf{G}_2 g, \\ \left[\frac{\lambda}{4\varepsilon}\mathbf{G}_0 + \lambda\varepsilon\mathbf{A}^\intercal\mathbf{G}_1\mathbf{A} + \mathbf{M}_{01}{}^\intercal\mathrm{diag}\left(\mathbf{B}'u\right)^2\mathbf{G}_1\mathbf{M}_{01}\right]v = \frac{\lambda}{4\varepsilon}\mathbf{G}_0 1. \end{cases}$$

Image restoration on toy examples

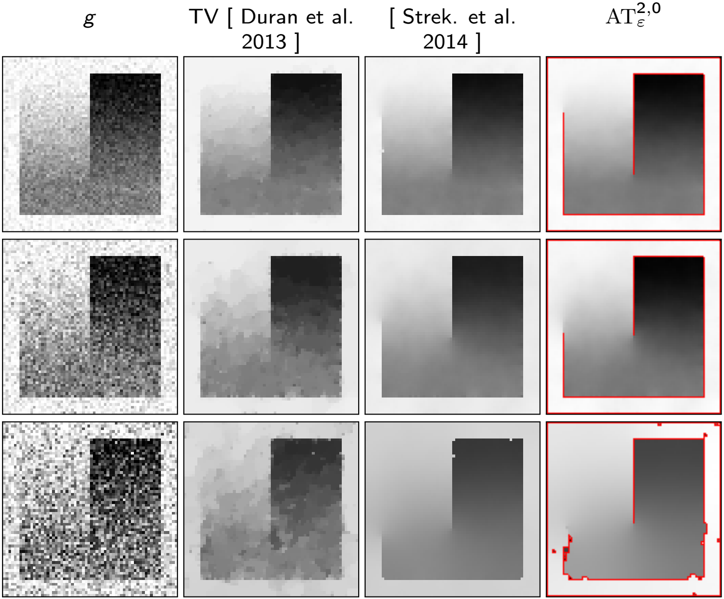


- Our algorithm progressively decreases ϵ to get a better chance of capturing the optimum
  - ▷ ϵ follows typically sequence $2, 1, 0.5, 0.25$ (for $h = 1$ sampling)
  - ▷ results on u and v are starting point for next ϵ
- systems are solved using Cholesky decomposition (Eigen)

Image restoration on toy examples



- Our algorithm progressively decreases ϵ to get a better chance of capturing the optimum
  - ▷ ϵ follows typically sequence $2, 1, 0.5, 0.25$ (for $h = 1$ sampling)
  - ▷ results on u and v are starting point for next ϵ
- systems are solved using Cholesky decomposition (Eigen)

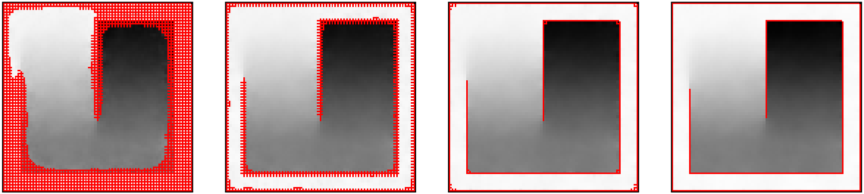


g
TV [Duran et al. 2013]
[Strek. et al. 2014]
$AT_{\varepsilon}^{2,0}$

Influence of parameter ε

$$AT_\varepsilon(u,v) = \alpha \int_\Omega |u-g|^2 \, \mathrm{dx} + \int_\Omega v^2 |\nabla u|^2 \, \mathrm{dx} + \lambda \int_\Omega \varepsilon |\nabla v|^2 + \frac{1}{\varepsilon} \frac{(1-v)^2}{4} \, \mathrm{dx}$$

- Γ-convergence parameter
- Controls the thickness of the contours
  - ▷ large ε convexifies AT and helps to detect the discontinuities;
  - ▷ as ε goes to 0, the discontinuities become thinner and thinner.

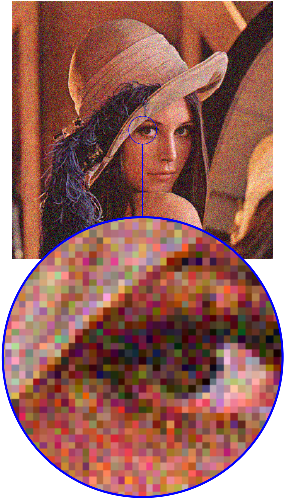


$\varepsilon = 2 \searrow 2$ | $\varepsilon = 2 \searrow 1$ | $\varepsilon = 2 \searrow 0.5$ | $\varepsilon = 2 \searrow 0.25$

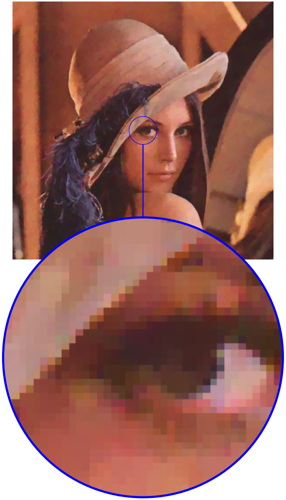
Discrete calculus model of Ambrosio-Tortorelli's functionnal

Image restoration / denoising

g
(PSNR = 20.23 dB)



TV
(PSNR = 29.36 dB)



$\mathrm{AT}_{\varepsilon}^{2,0}$
(PSNR = 29.03 dB)

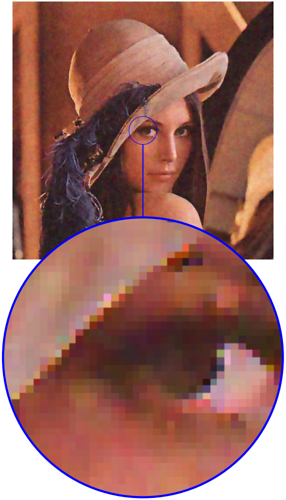
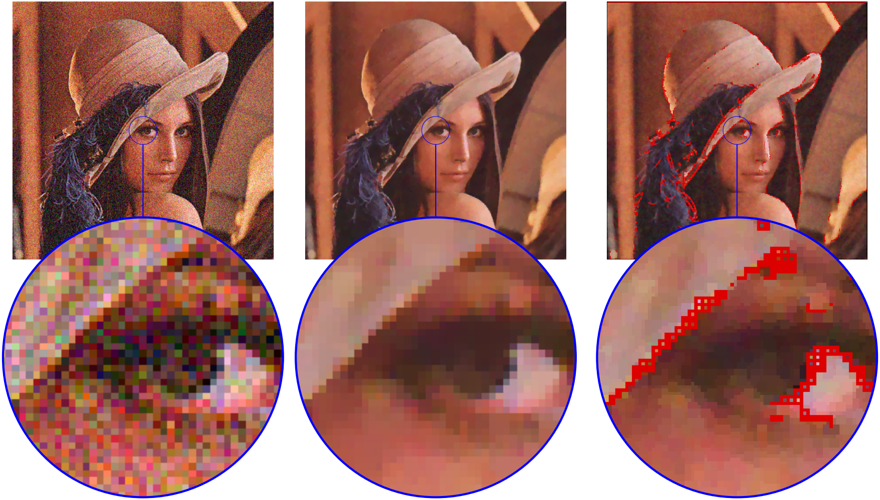


Image restoration / denoising

g	TV	$\mathrm{AT}_{\varepsilon}^{2,0}$
(PSNR = 20.23 dB)	(PSNR = 29.36 dB)	(PSNR = 29.03 dB)

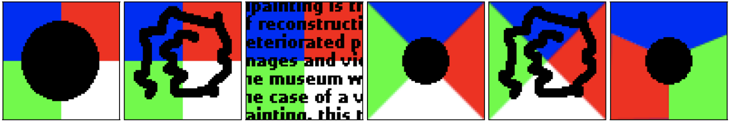


Scale-space given by α and λ and image segmentation

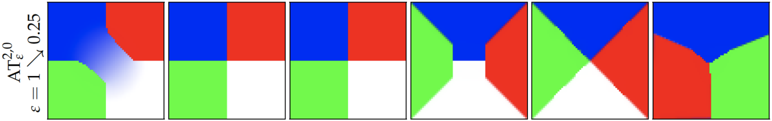
for decreasing λ

Image inpainting (on toy example)

- mask (in black) : domain M where data g (in color) is unknown
- $\alpha(x) := \{\alpha \in \Omega \setminus M, 0 \text{ elswhere}\}$
- initialization: u random in M, $= g$ in $\Omega \setminus M$



$\mathrm{AT}_\varepsilon^{2,0}$ with ϵ from **1** to 0.25



$\mathrm{AT}_\varepsilon^{2,0}$ with ϵ from **4** to 0.25

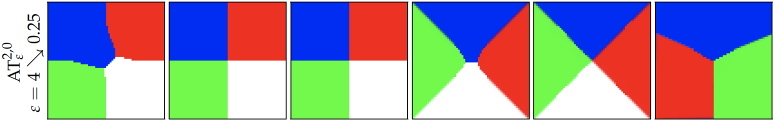
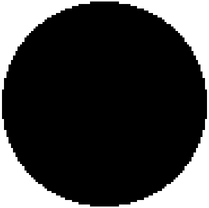
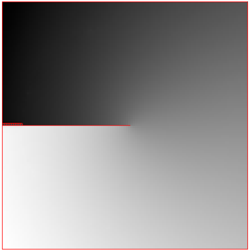


Image inpainting (on classical crack-tip example)

g



mask M

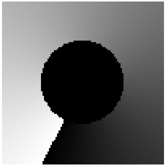


$\mathrm{AT}_{\varepsilon}^{2,0}$, $\alpha = 1$, $\lambda = 0.0024$

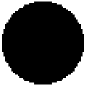
- Decreasing sequence of λ (irreversibility !?)
- same result as [Pock, Bishof, Cremers, Pock 2009], based on MS relaxation of [Alberti, Bouchitté, Dal Maso 2003]
- result independent of initialization as long as first ϵ is big enough (ϵ from 4 to 0.25 here, for image of size 110×110).

Image inpainting (crack-tip + decreasing λ)

g



mask M



$\alpha = 1$

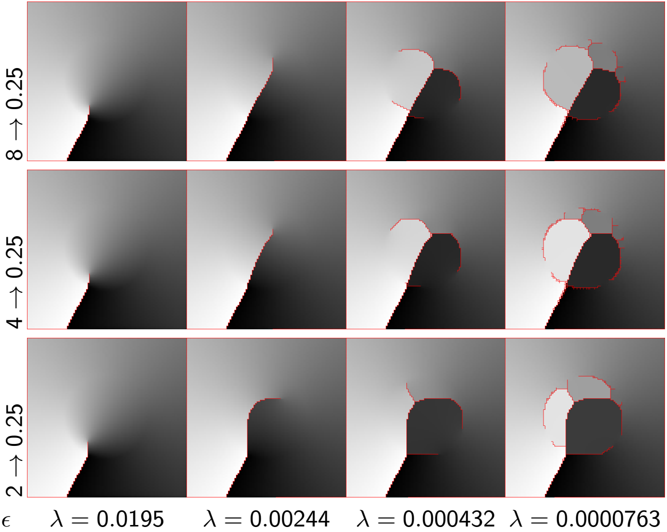
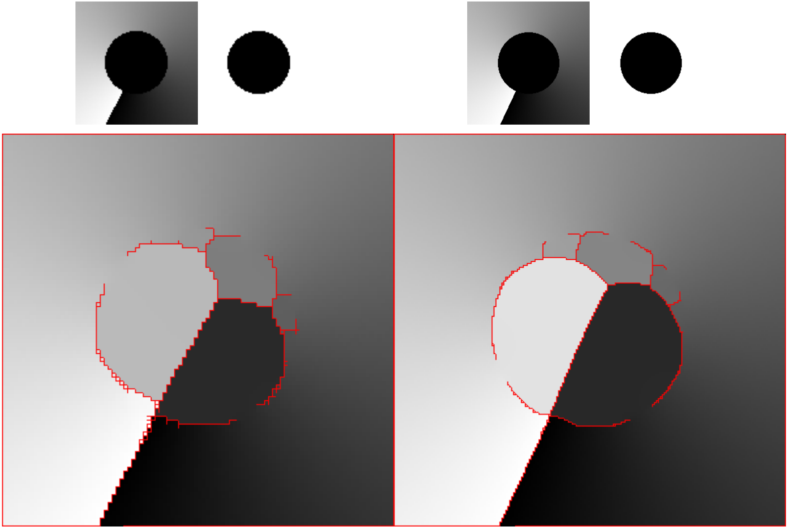


Image inpainting (crack-tip + changing resolution)

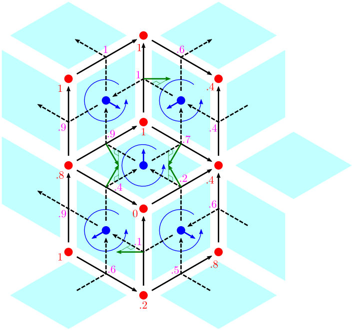


100×100
$\lambda = 0.0000763$
$\alpha = 1$

200×200
$\lambda = 0.0000381$
$\alpha = 1$

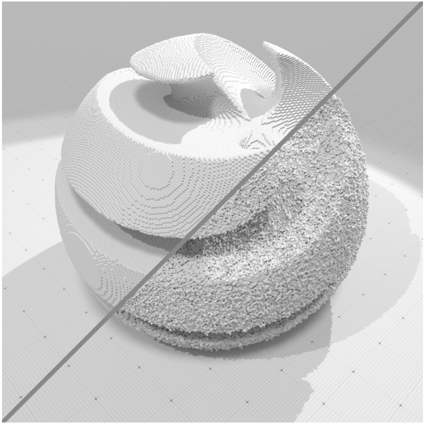
Feature delineation on digital surfaces

digital surface = boundary of set of voxels



same discrete calculus

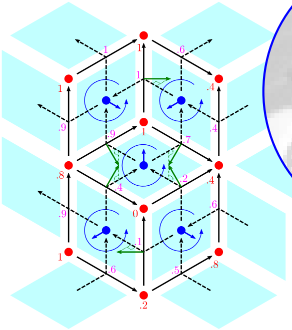
same $\mathrm{AT}_\varepsilon^{2,0}$



Input: normal vector field g estimated by Integral Invariant digital normal estimator.

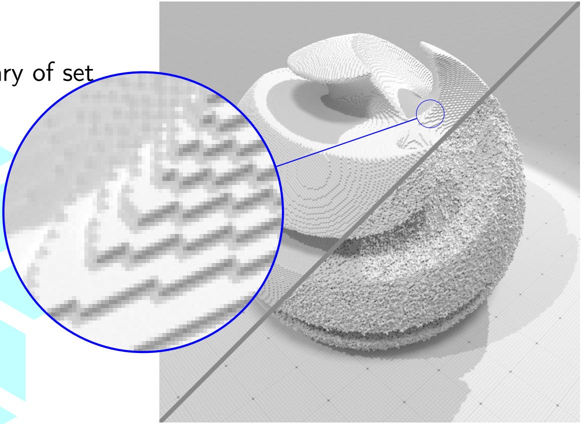
Feature delineation on digital surfaces

digital surface = boundary of set of voxels



same discrete calculus

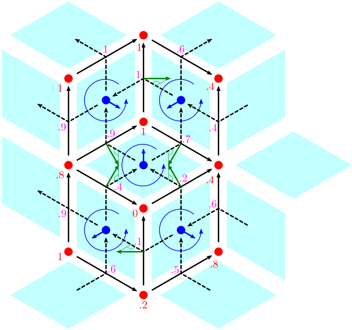
same $\mathrm{AT}_{\varepsilon}^{2,0}$



Input: normal vector field g estimated by Integral Invariant digital normal estimator.

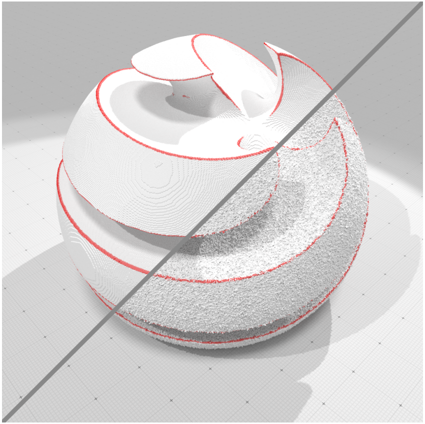
Feature delineation on digital surfaces

digital surface = boundary of set of voxels



same discrete calculus

same $\mathrm{AT}^{2,0}_{\varepsilon}$

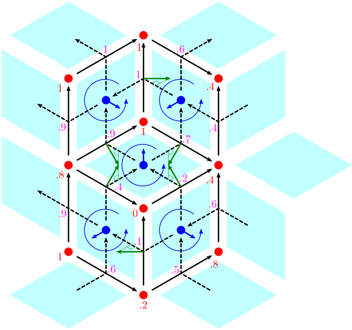


Input: normal vector field g estimated by Integral Invariant digital normal estimator.

Output: piecewise smooth normals $(u_i)_{i=1,2,3}$ and features v

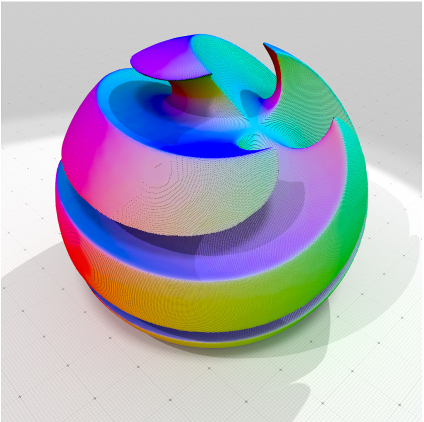
Feature delineation on digital surfaces

digital surface = boundary of set of voxels



same discrete calculus

same $\mathrm{AT}_{\varepsilon}^{2,0}$

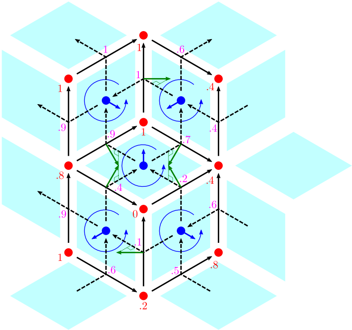


Input: normal vector field g estimated by Integral Invariant digital normal estimator.

Output: piecewise smooth normals $(u_i)_{i=1,2,3}$ and features v

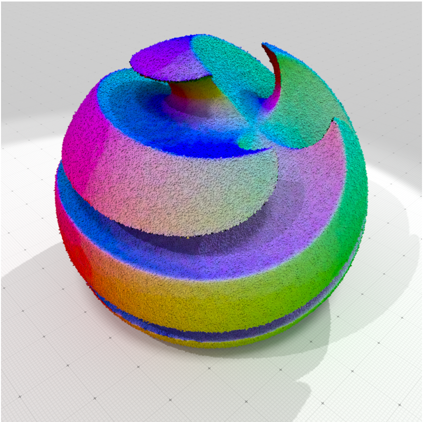
Feature delineation on digital surfaces

digital surface = boundary of set of voxels



same discrete calculus

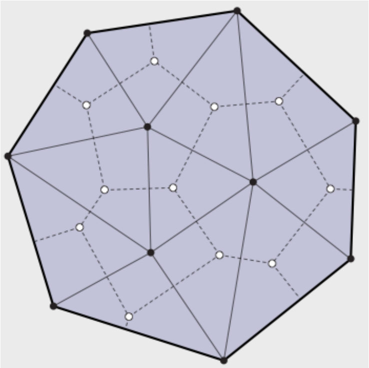
same $\mathrm{AT}_{\varepsilon}^{2,0}$



Input: normal vector field g estimated by Integral Invariant digital normal estimator.

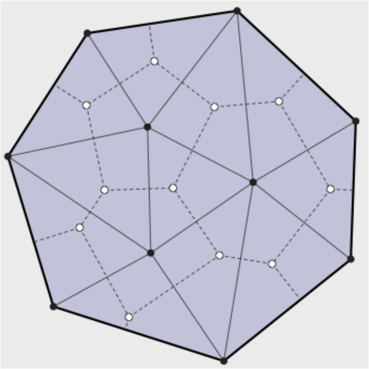
Output: piecewise smooth normals $(u_i)_{i=1,2,3}$ and features v

Discrete calculus on triangulated mesh



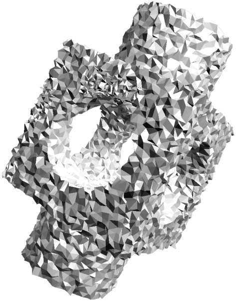
- dual mesh $\perp$ primal mesh
- dual vertex = center of triangle circumcircle
- Hodge stars are no more trivial but still diagonal matrices
- $\star_0(v) := \mathrm{Area}(\mathrm{dual}(v))$
- $\star_1(e) := \mathrm{length}(\mathrm{dual}(e))/\mathrm{length}(e)$
- $\star_2(t) := 1/\mathrm{Area}(t)$
- otherwise same discrete calculus

Discrete calculus on triangulated mesh



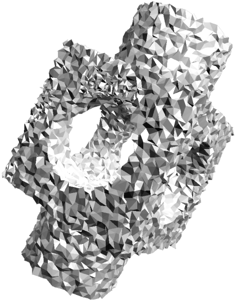
- dual mesh $\perp$ primal mesh
- dual vertex = center of triangle circumcircle
- Hodge stars are no more trivial but still diagonal matrices
- $\star_0(v) := \mathrm{Area}(\mathrm{dual}(v))$
- $\star_1(e) := \mathrm{length}(\mathrm{dual}(e))/\mathrm{length}(e)$
- $\star_2(t) := 1/\mathrm{Area}(t)$
- otherwise same discrete calculus
- $\mathrm{AT}_\varepsilon^{2,0}$ is then the same !

Mesh denoising



0. Bad mesh with positions $\mathbf{x}^0$, $k \leftarrow 0$

Mesh denoising

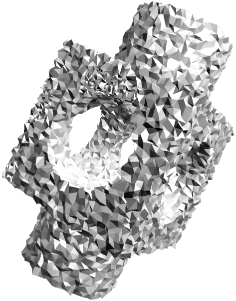


0. Bad mesh with positions $\mathbf{x}^0$, $k \leftarrow 0$

↓

1. g = normals from $\mathbf{x}^{(k)}$, Hodge stars from $\mathbf{x}^{(k)}$

Mesh denoising

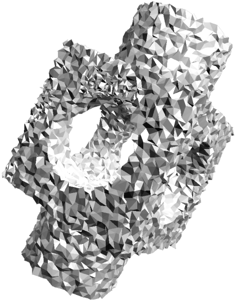


0. Bad mesh with positions $\mathbf{x}^0$, $k \leftarrow 0$

1. g = normals from $\mathbf{x}^{(k)}$, Hodge stars from $\mathbf{x}^{(k)}$

2. $\mathrm{AT}_\varepsilon^{2,0}$ to get piecewise smooth normals $u^{(k)}$

Mesh denoising



0. Bad mesh with positions $\mathbf{x}^0$, $k \leftarrow 0$

↓

1. g = normals from $\mathbf{x}^{(k)}$, Hodge stars from $\mathbf{x}^{(k)}$

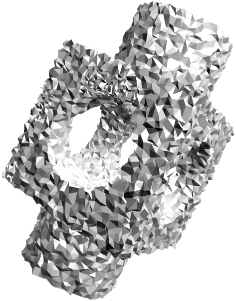
↓

2. $\mathrm{AT}_{\varepsilon}^{2,0}$ to get piecewise smooth normals $u^{(k)}$

↓

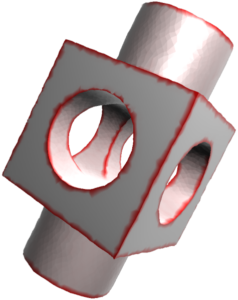
3. $\mathbf{x}^{(k+1)} \leftarrow$ regularize positions $\mathbf{x}^{(k)}$ by aligning geometric normals with $u^{(k)}$

Mesh denoising



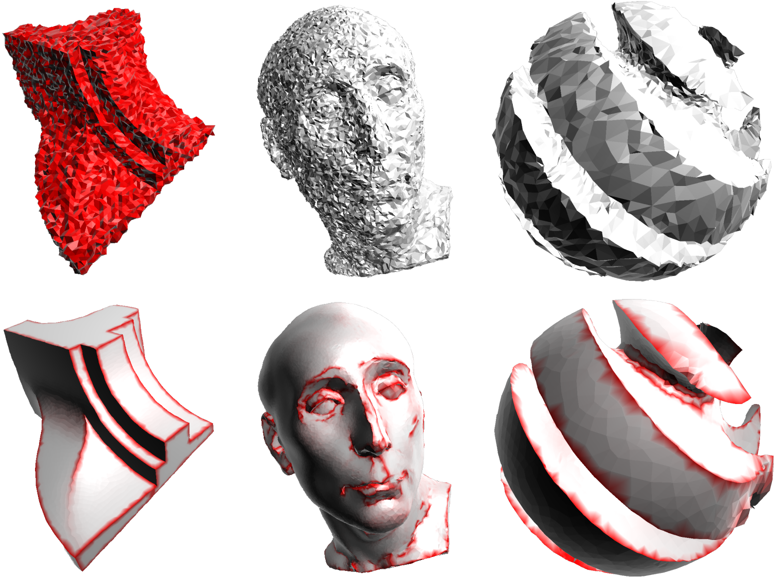
0. Bad mesh with positions $\mathbf{x}^0$, $k \leftarrow 0$
1. g = normals from $\mathbf{x}^{(k)}$, Hodge stars from $\mathbf{x}^{(k)}$
2. $\mathrm{AT}_{\varepsilon}^{2,0}$ to get piecewise smooth normals $u^{(k)}$
3. $\mathbf{x}^{(k+1)} \leftarrow$ regularize positions $\mathbf{x}^{(k)}$ by aligning geometric normals with $u^{(k)}$
4. $k \leftarrow k+1$ and iterate till stability (return to step 1)

Mesh denoising

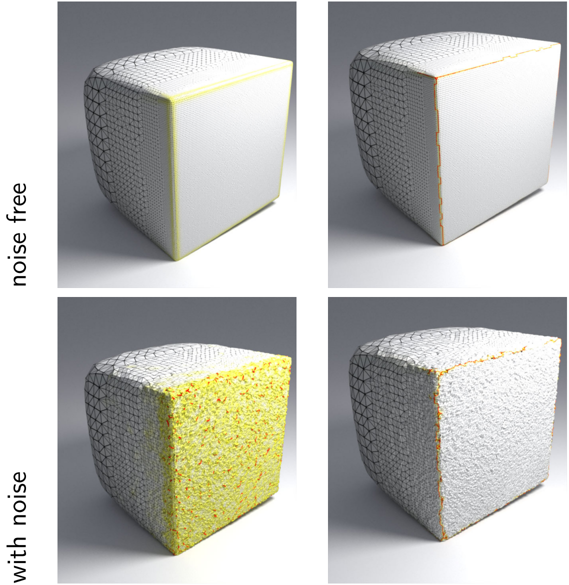


0. Bad mesh with positions $\mathbf{x}^0$, $k \leftarrow 0$
1. g = normals from $\mathbf{x}^{(k)}$, Hodge stars from $\mathbf{x}^{(k)}$
2. $\mathrm{AT}_{\varepsilon}^{2,0}$ to get piecewise smooth normals $u^{(k)}$
3. $\mathbf{x}^{(k+1)} \leftarrow$ regularize positions $\mathbf{x}^{(k)}$ by aligning geometric normals with $u^{(k)}$
4. $k \leftarrow k+1$ and iterate till stability

Mesh denoising (a few results)

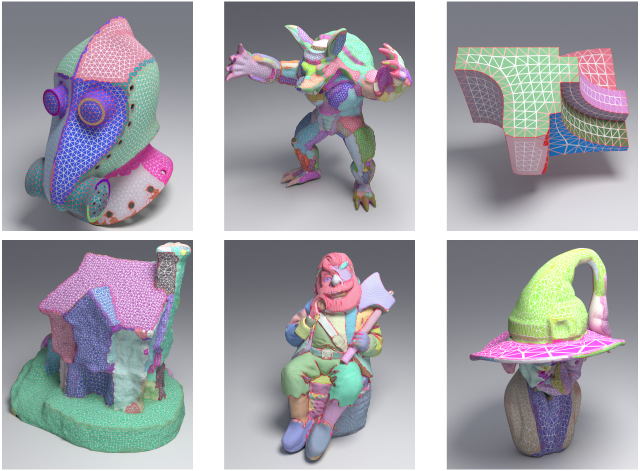


Mesh denoising (Comparison with FEM)



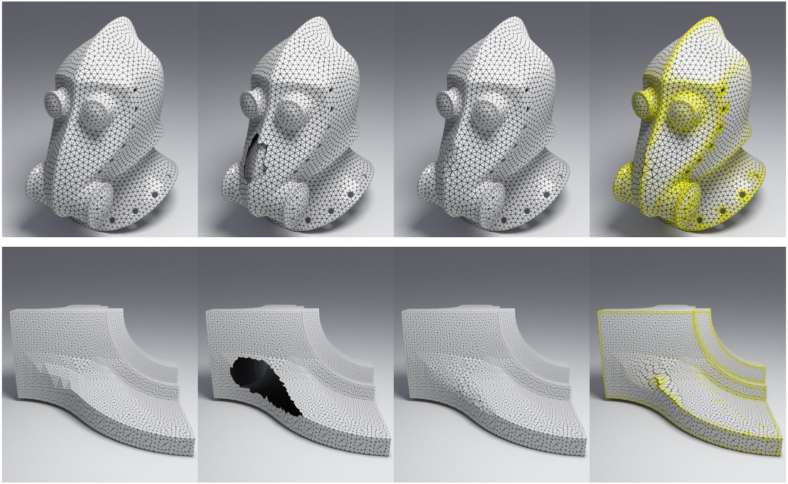
FEM [Tong and Tai 2016]   our approach

Mesh segmentation



- v is used as a probability of edge merge in a graph connected component algorithm

Mesh inpainting



Original | Missing area | CGAL filling | Our inpainting

Conclusion

- Discrete calculus model of AT recovers discontinuities
  - usual "phase-field" ones $\longrightarrow$ **thin** discontinuities
- very generic formulation: 2D images, digital surfaces, triangulated meshes, graph structures, 3D hexahedral, tetrahedral or mixed meshes, ...
- opens a wide range of applications
  - image processing
  - 3D geometry processing
- open-source C++ code available, mostly on `dgtal.org`, otherwise on `github.com`
- reasonnable computation times: from seconds to a few minutes

References I

Alberti, G., Bouchitté, G., and Dal Maso, G. (2003).
The calibration method for the mumford-shah functional and free-discontinuity problems.
Calculus of Variations and Partial Differential Equations, 16(3):299–333.

Ambrosio, L. and Tortorelli, V. M. (1992).
On the approximation of free discontinuity problems.
Boll. Un. Mat. Ital., 6(B):105–123.

Bourdin, B. and Chambolle, A. (2000).
Implementation of an adaptive finite-element approximation of the mumford-shah functional.
Numerische Mathematik, 85(4):609–646.

Boykov, Y. and Funka-Lea, G. (2006).
Graph cuts and efficient nd image segmentation.
International journal of computer vision, 70(2):109–131.

Boykov, Y., Veksler, O., and Zabih, R. (2001).
Fast approximate energy minimization via graph cuts.
IEEE Trans. Pattern Analysis and Machine Intelligence, 23(11):1222–1239.

Chambolle, A. and Pock, T. (2011).
A first-order primal-dual algorithm for convex problems with applications to imaging.
Journal of Mathematical Imaging and Vision, 40(1):120–145.

References II

Kee, Y. and Kim, J. (2014).
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*, pages 4074–4081.

Mumford, D. and Shah, J. (1989).
Optimal approximations by piecewise smooth functions and associated variational problems.
Communications on pure and applied mathematics, 42(5):577–685.

Pock, T., Cremers, D., Bischof, H., and Chambolle, A. (2009).
An algorithm for minimizing the mumford-shah functional.
In *2009 IEEE 12th International Conference on Computer Vision*, pages 1133–1140. IEEE.

Rudin, L. I., Osher, S., and Fatemi, E. (1992).
Nonlinear total variation based noise removal algorithms.
Physica D: Nonlinear Phenomena, 60(1):259–268.

Vese, L. A. and Chan, T. F. (2002).
A multiphase level set framework for image segmentation using the mumford and shah model.
International journal of computer vision, 50(3):271–293.