

# Fast algorithms for discontinuity-preserving smoothing

Martin Storath

*Technische Hochschule Würzburg-Schweinfurt*

DIPOpt Workshop, Lyon, 27.11.2023

# Some signals and images with discontinuities/jumps/steps

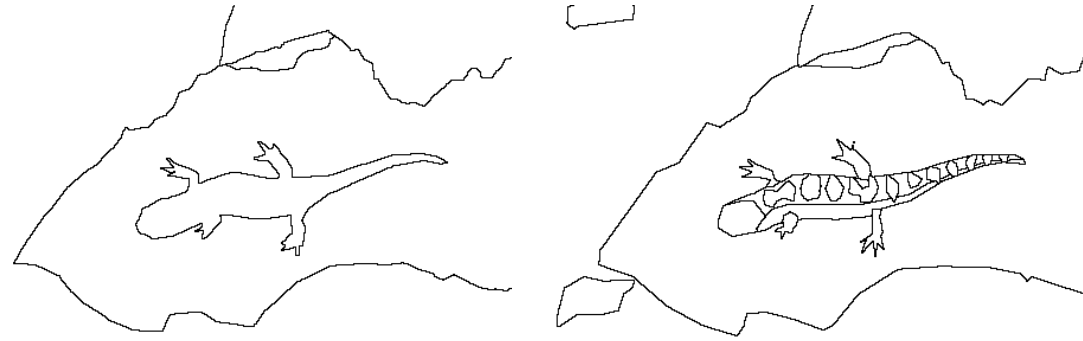
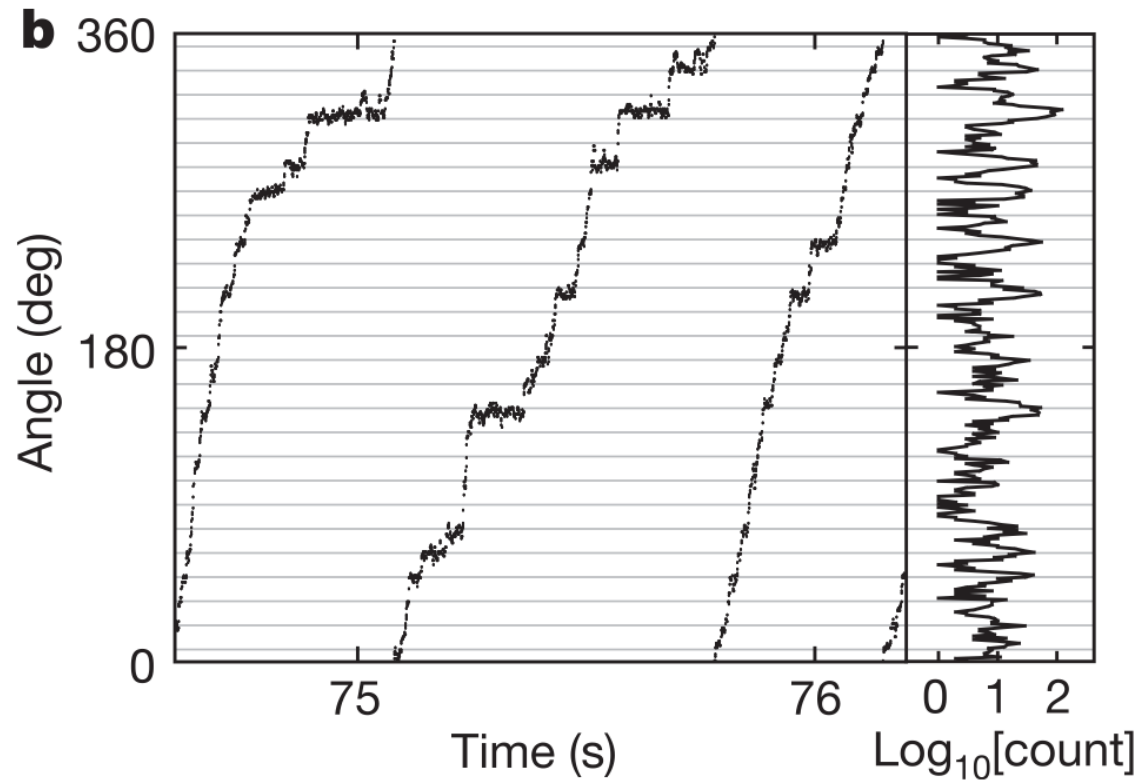
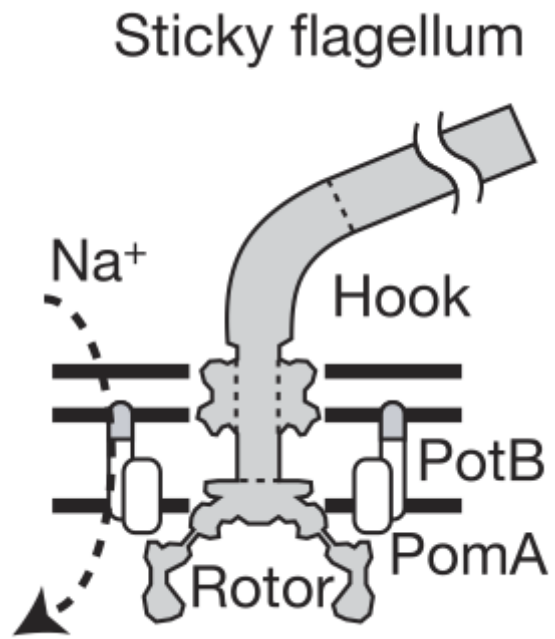


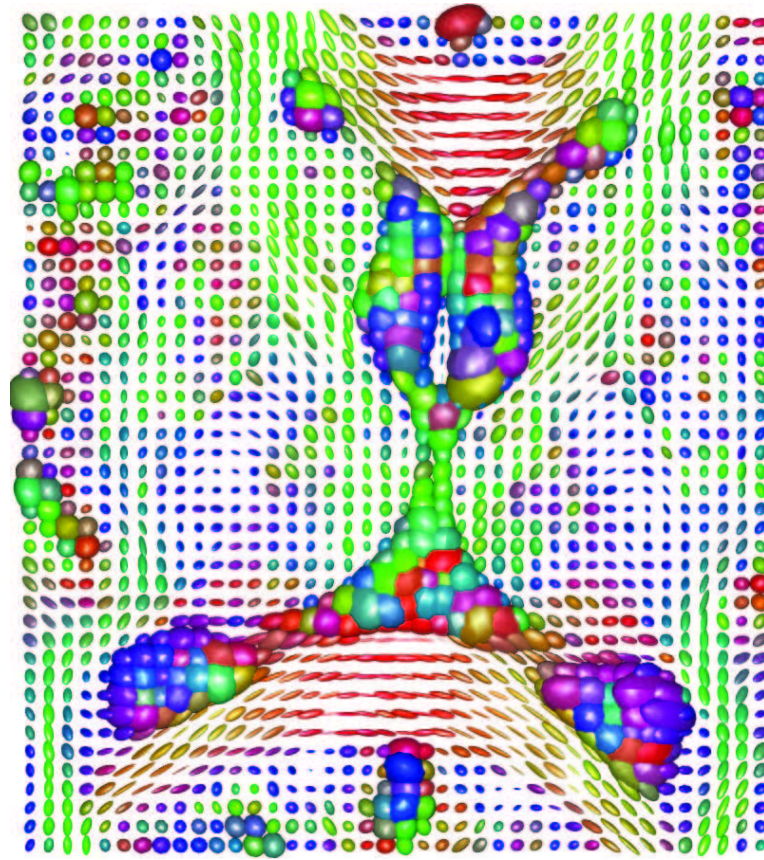
Image and two human segmentations from the Berkeley image segmentation data set

(Martin et al., ICCV 2001, Arbelaez et al. IEEE TPAMI, 2011)



Direct observation of steps in rotation of the bacterial flagellar

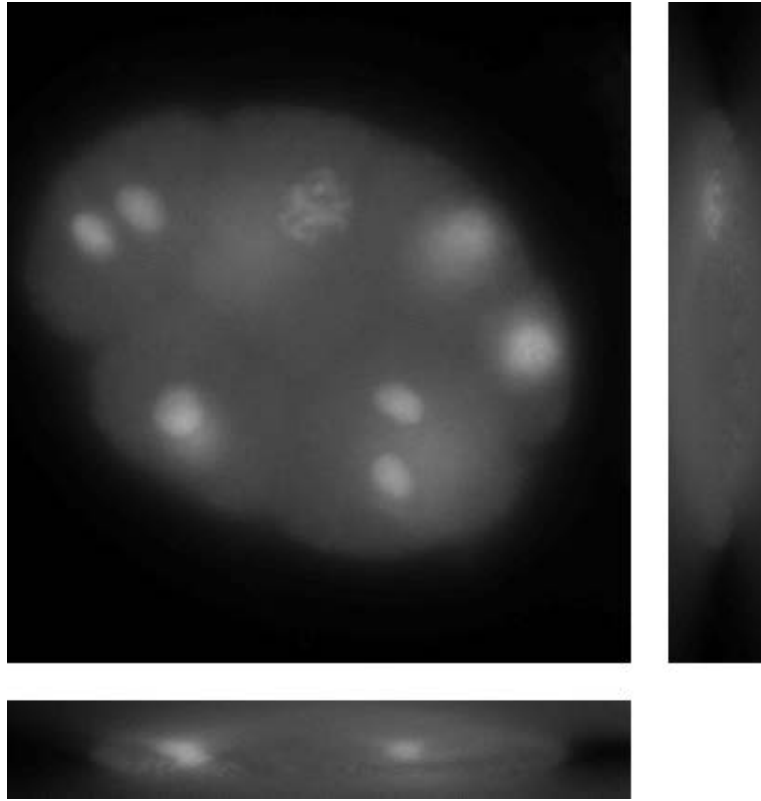
(Sowa et al., Nature 2005)



Diffusion Tensor Image of the human brain

(Baust et al., IEEE TMI, 2016; Data from Camino project)





Widefield microscopy image of a C-Elegans embryo  
(Sage et al., Methods, 2017; Storath et al., IEEE TIP, 2017)

# Fast algorithms for discontinuity-preserving smoothing: Contents

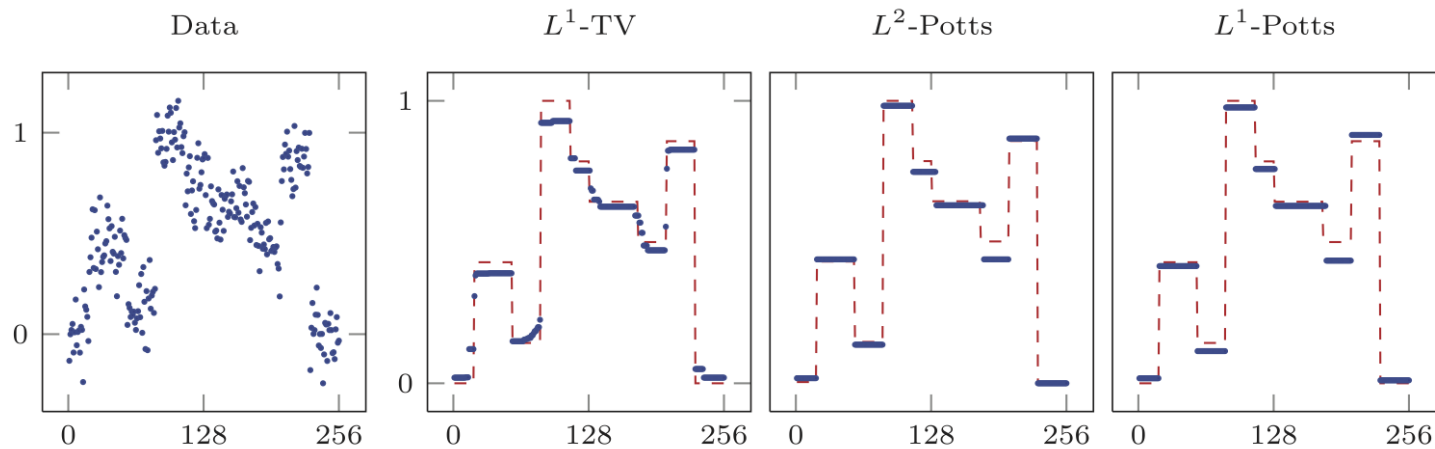
- The 1D Case
  - $L^1$ -Potts model for robust piecewise constant regression
  - Potts model for circle valued data
  - Cubic smoothing splines with discontinuities (CSSD)
- The 2D Case
  - Potts model for unsupervised image segmentation
  - Piecewise affine models and piecewise smooth models
  - Discretizations
  - Taylor jet based solvers

# Piecewise constant regression

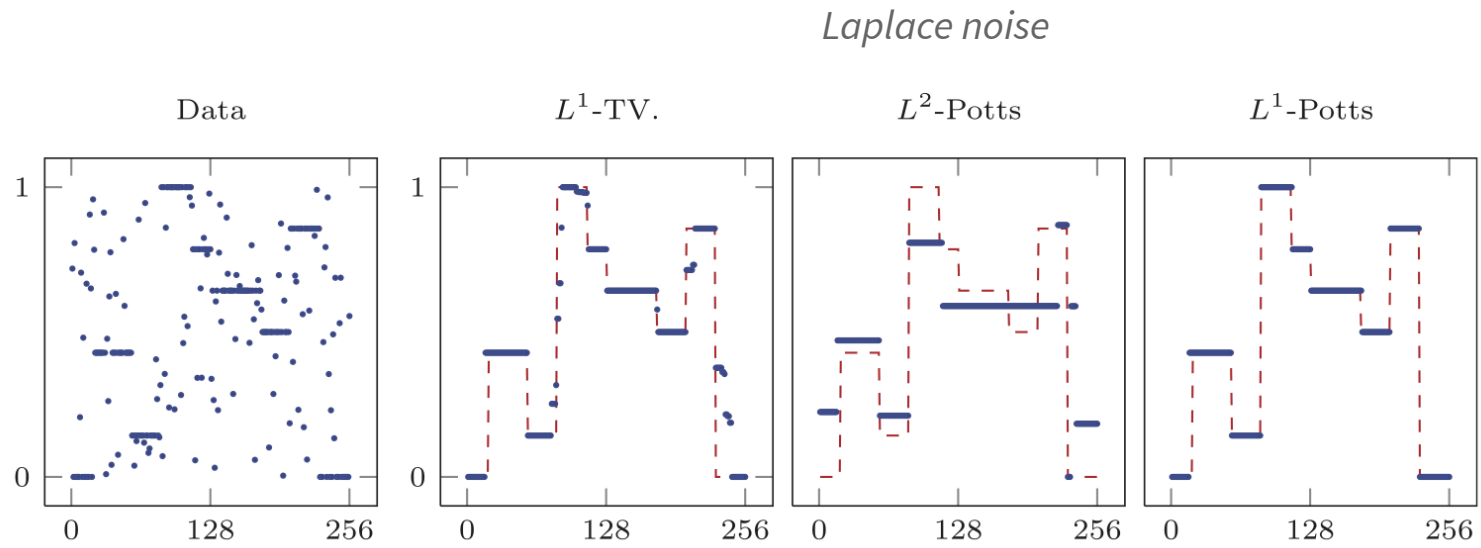
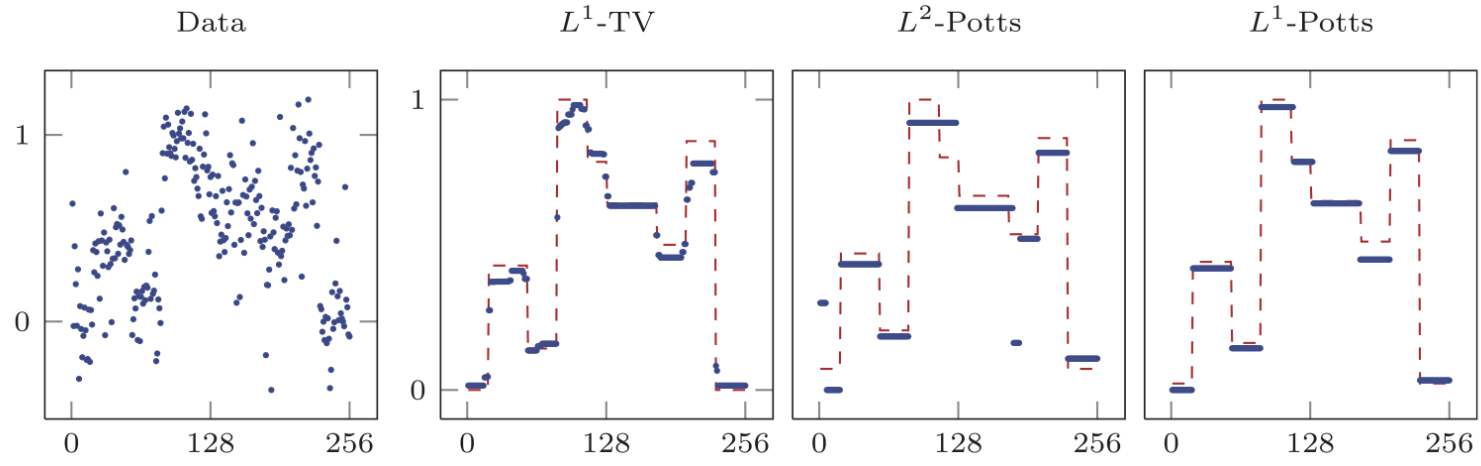
## Potts model

$$u^* \in \arg \min_{u \in \mathbb{R}^N} \gamma \|\nabla u\|_0 + \|u - f\|_p^p$$

- Data  $f \in \mathbb{R}^N$ ,  $\|\nabla u\|_0 = |\{i : u_i \neq u_{i+1}\}|$  number of jumps, non-convex
- Related to total variation  $\|\nabla u\|_1 = \sum_i |u_i - u_{i+1}|$ , convex



*Piecewise constant signal with Gaussian noise*



*Salt and pepper noise*



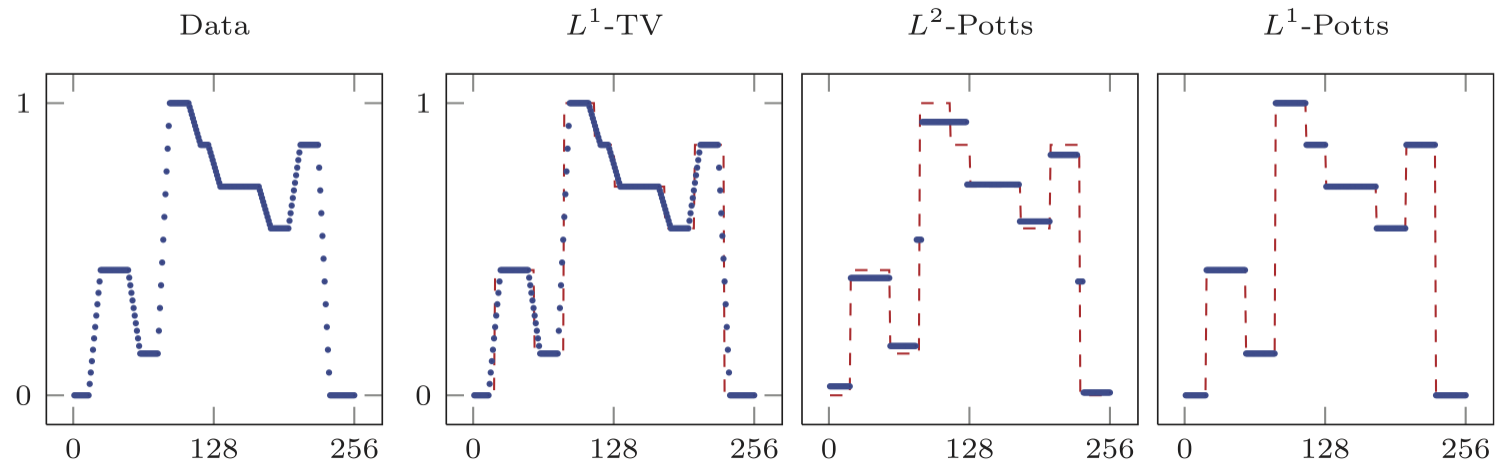
# Blind deconvolution property of $L^1$ -Potts

Assume that  $K$  is symmetric, compactly supported, nonnegative with  $K > 0$  in a neighborhood of the origin, and that it has total mass 1.

**Theorem (Weinmann/S./Demaret, SINUM 2015)**

If  $g$  is piecewise constant and the support size of  $K$  is sufficiently small then we find a range of  $\gamma$ -values such that

$$g = \arg \min_{u \in \mathbb{R}^N} \gamma \|\nabla u\|_0 + \|u - K * g\|_1.$$



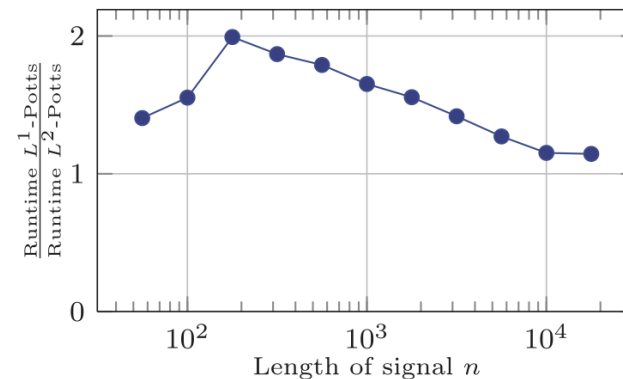
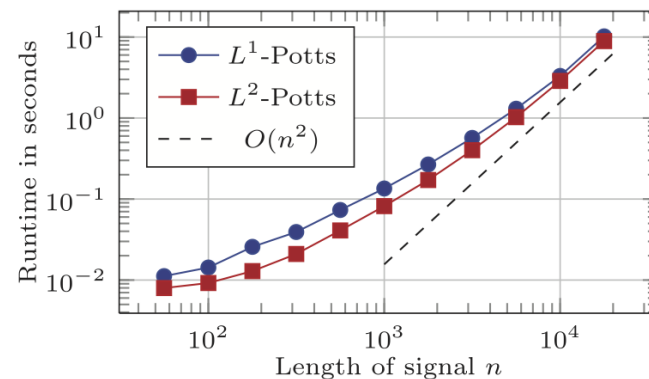
# Fast algorithms of the 1D Potts problem

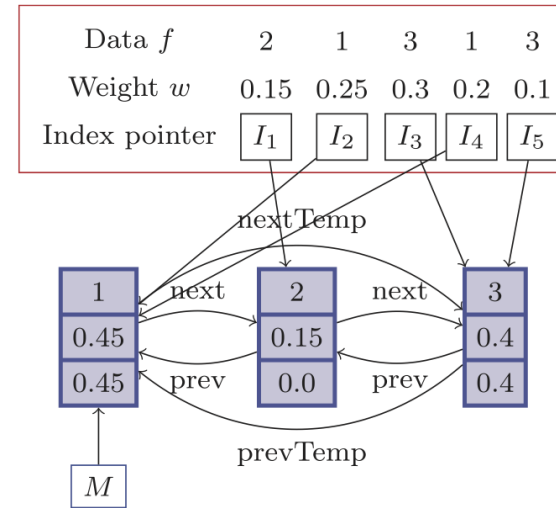
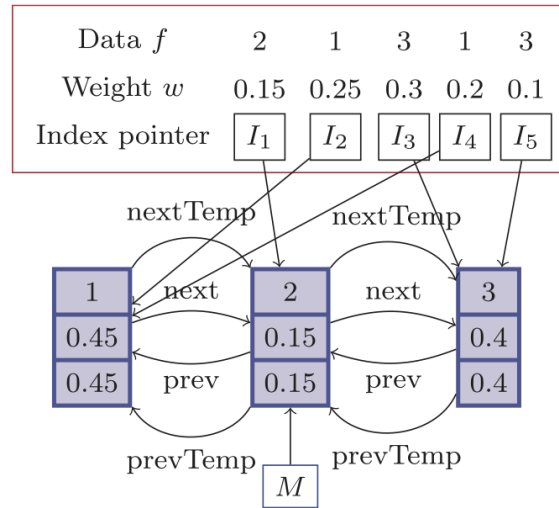
General approach for 1D Potts problems (Winkler/Liebscher 2002)

- by dynamic programming only  $O(N^2)$  configurations to be checked
- need value  $\min_{\mu \in \mathbb{R}} \sum_{n=l}^r |\mu - f_n|^p$  for all discrete intervals  $l, l+1, \dots, r$

Time complexity:

- for  $L^2$ -Potts:  $O(N^2)$  (Winkler/Liebscher 2002) and  $O(N)$  if the number of jumps grows linearly (Killick et al. 2012)
- for  $L^1$ -Potts:  $O(N^2)$  (Weinmann/S./Demaret 2015)





*Absolute deviation update scheme for fast  $L^1$ -Potts solver*



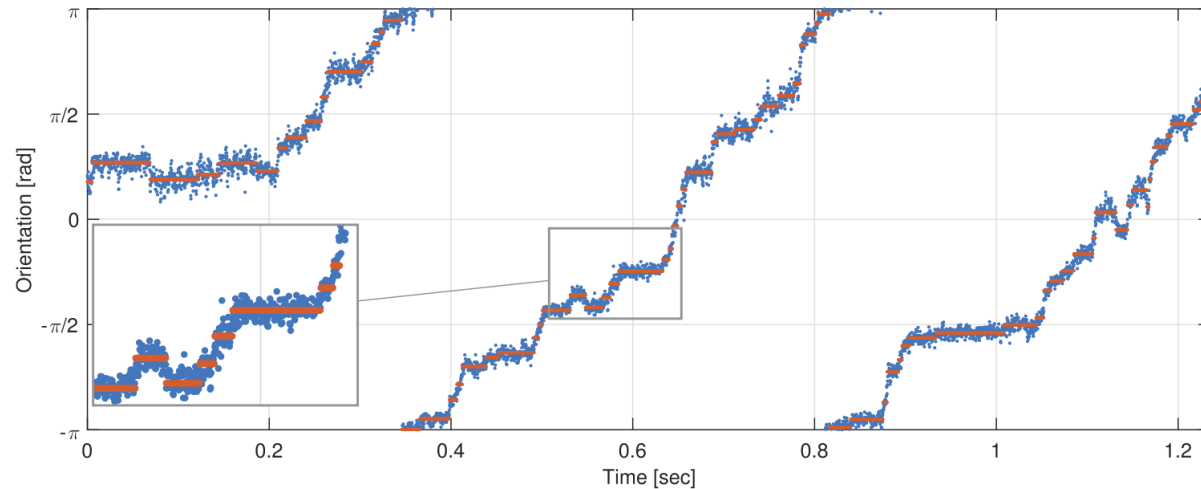


# Circle-valued data

$L^1$ -Potts model for  $f \in \mathbb{T}^N$

$$\arg \min_{u \in \mathbb{R}^N} \gamma \|\nabla u\|_0 + \sum_{n=1}^N w_n d(u_n, f_n)$$

where  $d(x, y)$  the arc length between two points on the unit circle, and  $w \in \mathbb{R}^N$ .



*Time series of angles of the bacterial flagellar motor and  $L^1$ -Potts model on the circle.  
(S./Weinmann/Unser, 2017; Data courtesy of Sowa et al.)*

# Fast algorithm for the Potts problem with circle-valued data

Challenge: fast histogram update of real-valued problem cannot be used directly

## Proposed approach (S./Weinmann/Unser 2017)

1. May reduce search space to the finite set  $V = \{f_n | n = 1, \dots, N\}$  so that

$$\arg \min_{u \in V^N} \gamma \|\nabla u\|_0 + \sum_{n=1}^N w_n d(u_n, f_n)$$

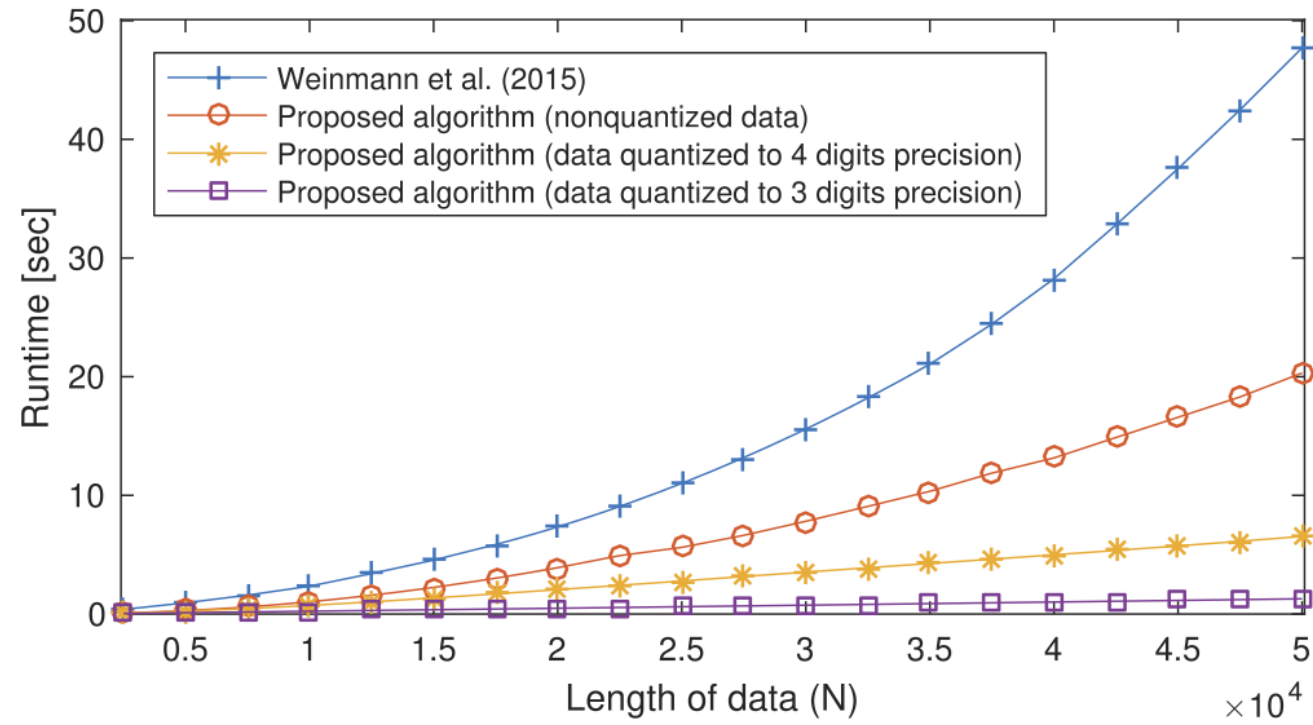
2. Use a Viterbi-type algorithm: Set  $B_k^1 = w_1 d(v_k, y_1)$  for  $k = 1, \dots, K$  and

$$B_k^n = w_n d(v_k, y_n) + \min_l \{B_l^{n-1} + \gamma \mathbf{1}_{v_k \neq v_l}\}, \text{ for } k = 1, \dots, K, n = 1, \dots, N.$$

3. Can save a factor  $K$  by a result of Felzenszwalb/Huttenlocher (2006)

4. The optimal solution is obtained via backtracking.

Resulting time complexity:  $O(NK)$  where  $K = |V| \leq N$



# Further aspects of 1D Potts models

## Hyperparameter selection

- General methods: AIC, BIC, crossvalidation
- Modified BIC (Zhang 2007)
- Bayesian approach (Frecon et al. 2017)

## Extensions

- Extension to vector-valued data
  - straightforward in  $L^2$  case
  - more challenging for  $L^1$  case for vector median
- Potts penalty on slope for piecewise linear estimation (Fearnhead et al. 2017)
- “Inverse” Potts problem for indirectly measured data
$$u^* \in \arg \min_{u \in \mathbb{R}^N} \gamma \|\nabla u\|_0 + \|Au - f\|_p^p$$
  - Splitting methods (S. et al, 2014)
  - Iterative method based on surrogate functionals (Weinmann/S. 2015)
- Extension to manifold-valued data possible (Weinmann et al. 2016)

# Piecewise smooth models

Reminder: **Smoothing splines** are solutions of

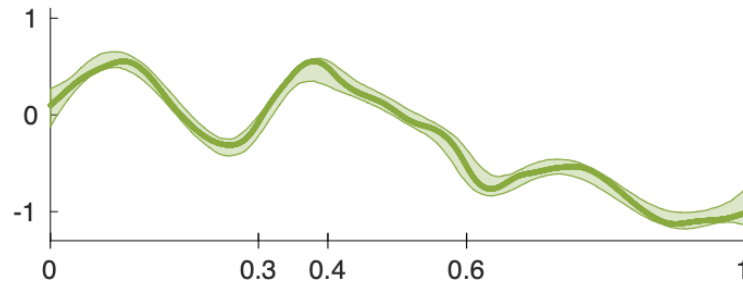
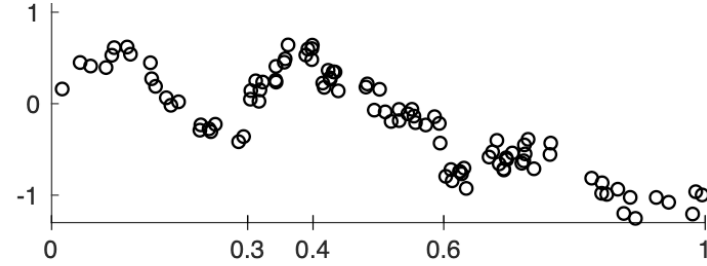
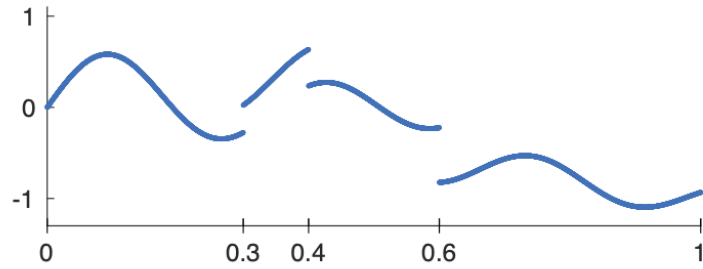
$$\min_{f \in C^2} p \sum_{i=1}^N \left( \frac{y_i - f(x_i)}{\delta_i} \right)^2 + (1 - p) \int_{x_1}^{x_N} (f''(t))^2 dt$$

- non-parametric regressors with adjustable degree of smoothness  $0 < p < 1$
- $y \in \mathbb{R}^N$  measured data at data sites  $x \in \mathbb{R}^N$
- $\delta \in \mathbb{R}^N$  standard deviations of the data sites

Popular tool for smoothing/denoising signals and time series because

- they provide flexible approximation,
- are computationally efficient,
- give smooth solutions.

## ...but splines smooth out discontinuities



*Ground truth, noisy samples, and smoothing spline with  $p=0.999$   
Shaded area: 95 % quantiles over 1000 realizations*

# Smoothing splines with discontinuities

Piecewise smoothing spline model

$$\min_{f, J} p \sum_{i=1}^N \left( \frac{y_i - f(x_i)}{\delta_i} \right)^2 + (1 - p) \int_{[x_1, x_N] \setminus J} (f''(t))^2, dt + \gamma |J|.$$

- penalty on the number of jumps  $|J|$  weighted by  $\gamma > 0$ .
- Solutions of this optimization problems are **cubic smoothing splines with discontinuities (CSSD)**.

Questions:

- Are the results unique?
- How to solve the non-convex optimization problem accurately and efficiently?
- How to choose the hyperparameters  $p$  and  $\gamma$ ?



## Relations to other models

- CSSD model is a special case of the weak rod model (Blake/Zisserman 1987) without creases and in semidiscrete setup
- and a second order variant of the Mumford-Shah model

## Prior and related algorithms

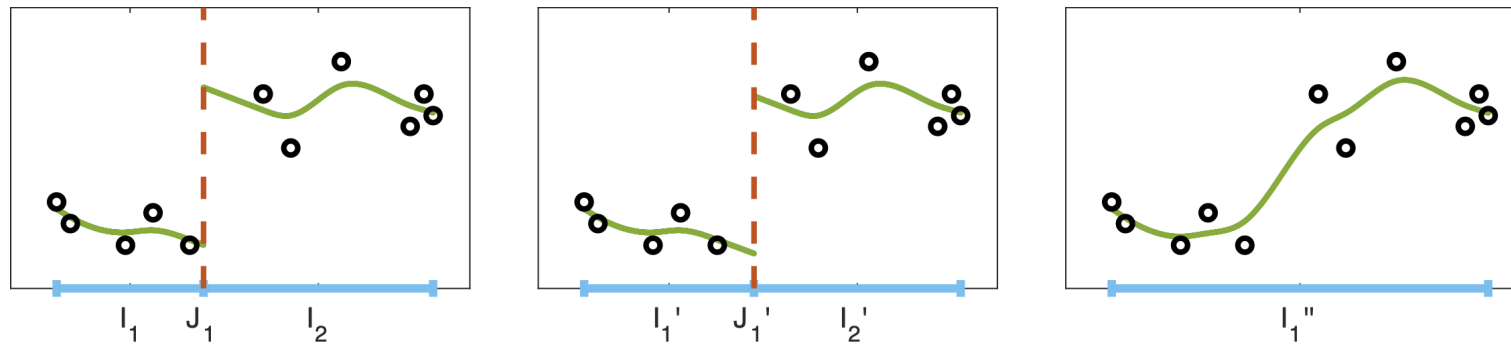
- linear time algorithm for smoothing splines (Reinsch 1967)
- combination of the above method with dynamic program of Liebscher/Winkler gives an exact solver for the CSSD problem with complexity  $O(N^3)$  (baseline)
- non-exact solvers for weak rod model: graduated non-convexity, Hopfield's neural model, Viterbi-type algorithm (Blake/Zisserman 1987; Blake, 1989)
- algorithm for a fully discrete spline with equidistant samples (S./Kiefer/Weinmann 2019) with complexity  $O(N^2)$

**Next:** Fast algorithm for continuous formulation with non-equidistant samples (S./Weinmann, 2023)

# Formulation as partitioning problem

Discontinuity locations  $J$  induce a partition on the domain  $[x_1, x_N]$ :

$$[x_1, x_N] \setminus J = I_1 \cup \dots \cup I_{K+1} \text{ and } \mathcal{P}(J) = I_1, \dots, I_{K+1}.$$

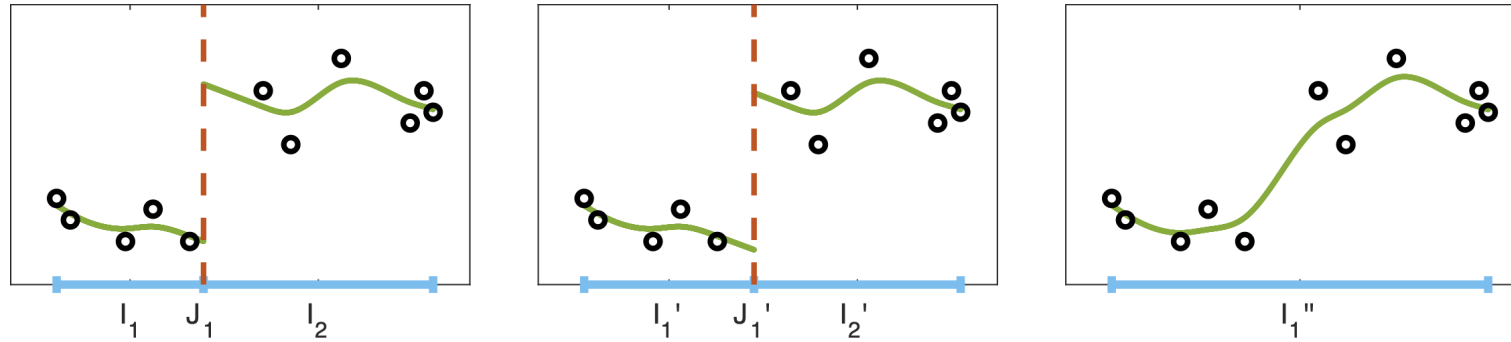


Reformulation of the CSSD model as

$$\min_{J \subset [x_1, x_N] \setminus \{x_1, \dots, x_N\}} \sum_{I \in \mathcal{P}(J)} \mathcal{E}_I + \gamma |J|$$

$$\mathcal{E}_I = \min_{f \in C^2(I)} p \sum_{i: x_i \in I} \left( \frac{y_i - f(x_i)}{\delta_i} \right)^2 + (1 - p) \int_I (f''(t))^2 dt.$$

# Reducing the search space



- Shift of jump locations between to data sites does not affect the target value
- Reduce the search space for  $J$  to set of midpoints of the data sites  $\mathcal{M}$

Obtain the semi-discrete optimization problem

$$\min_{J \subset \mathcal{M}} \sum_{I \in \mathcal{P}(J)} \mathcal{E}_I + \gamma |J|.$$

# Uniqueness result

Apart from the shift in between two data points we obtain uniqueness.

## Theorem (S./Weinmann, JCGS 2023)

Minimizing partitions with all segments containing at least three data points and corresponding minimizing functions are uniquely determined for (Lebesgue)-a.e. input  $y \in \mathbb{R}^N$ .

Concept of proof:

- Show that data that yield non-unique partitions are contained in a union of levelsets of non-trivial quadratic forms which are a Lebesgue zero set.
- Smoothing splines on each interval of a partition are uniquely determined

# Fast exact algorithm

- Basis: dynamic programming approach of Winkler/Liebscher (2002)
- **Key new element:** update strategy for computing

$$\mathcal{E}_I = \min_{f \in C^2(I)} p \sum_{i: x_i \in I} \left( \frac{y_i - f(x_i)}{\delta_i} \right)^2 + (1 - p) \int_I (f''(t))^2 dt.$$

in  $O(1)$  for each interval  $I$ .

## Theorem (S./Weinmann, JCGS 2023)

The proposed algorithm computes a solution of the CSSD model. The worst case time complexity is  $O(N^2)$  and the memory complexity is  $O(N)$ .

- Speed-up possible by pruning strategies (Killick et al. 2012; S./Weinmann 2014)
- Implementation at <https://github.com/mstorath/CSSD>

## Details on the fast algorithm

Define  $A^{(r)} \in \mathbb{R}^{2r \times s}$ , with  $s = 3r - 2$ , by

$$A^{(r)} = \begin{bmatrix} \alpha_1 e_1^T & 0 & 0 & \cdots & 0 \\ \beta V_1 & \beta W_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 e_1^T & 0 & \cdots & 0 \\ 0 & \beta V_2 & \beta W_2 & \cdots & 0 \\ 0 & 0 & \alpha_3 e_1^T & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \beta V_{r-1} & \beta W_{r-1} \\ 0 & \cdots & 0 & 0 & \alpha_r e_1^T \end{bmatrix}$$

with  $\alpha_i = \frac{\sqrt{p}}{\delta_i}$ ,  $\beta = \sqrt{1-p}$ ,  $d_i = x_{i+1} - x_i$ ,

$$[V_i, U_i] = \begin{bmatrix} \frac{2\sqrt{3}}{d_i^{3/2}} & \frac{\sqrt{3}}{\sqrt{d_i}} & -\frac{2\sqrt{3}}{d_i^{3/2}} & \frac{\sqrt{3}}{\sqrt{d_i}} \\ 0 & \frac{1}{\sqrt{d_i}} & 0 & -\frac{1}{\sqrt{d_i}} \end{bmatrix}$$

### Theorem (S./Weinmann, JCGS 2023)

The spline approximation error in the interval  $[x_1, x_r]$  has the form

$$\mathcal{E}_{\{1:r\}} = \min_{u \in \mathbb{R}^{2r}} \|A^{(r)}u - \tilde{y}^{(r)}\|_2^2,$$

where  $\tilde{y}^{(r)} \in \mathbb{R}^s$  is a vector of zeros except  $\tilde{y}_{3i-2}^{(r)} = \alpha_i y_i$  for  $i = 1, \dots, r$ .





## Key observations for the update step

- $A^{(r)}$  is a submatrix of  $A^{(r+1)}$ , and  $\tilde{y}^{(r)}$  is a subvector of  $\tilde{y}^{(r+1)}$ .
- QR-decomposition of the system  $[A^{(r)} | \tilde{y}^{(r)}]$  can be updated to a QR-decomposition of the system  $[A^{(r+1)} | \tilde{y}^{(r+1)}]$  by a constant number of Givens rotations
- The Givens rotations act only on a subsystem  $[R|z]$  with  $R \in \mathbb{R}^{5 \times 4}$
- The approximation error has the update recurrence

$$\mathcal{E}_{\{1:r+1\}} = \mathcal{E}_{\{1:r\}} + (z'_5)^2$$

where  $[R'|z']$  is the system after the Givens rotations.

### Theorem (S./Weinmann, JCGS 2023)

The procedure described above computes the array  $[\mathcal{E}_{\{1:1\}}, \mathcal{E}_{\{1:2\}}, \dots, \mathcal{E}_{\{1:N\}}]$  in  $O(N)$  time complexity and  $O(N)$  memory complexity.



# Comparison of computational effort

- Baseline: standard computation of the spline approximation error using the PELT method (Python package ruptures)
- proposed method with PELT pruning (Killick et al. 2012)
- proposed method with FPVI pruning (S./Weinmann 2014)

Signal length $N$	250	500	1000	2000	4000	8000
<b>Constant number of discontinuities</b>						
Baseline	$1.1 \cdot 10^6$	$2.7 \cdot 10^6$	$2.0 \cdot 10^7$	$1.5 \cdot 10^8$	$1.3 \cdot 10^9$	$7.3 \cdot 10^9$
	45.7	110.1	453.4	1909.1	8704.0	33331.3
Prop. + FPVI	$2.5 \cdot 10^4$	$10.0 \cdot 10^4$	$4.0 \cdot 10^5$	$1.7 \cdot 10^6$	$7.0 \cdot 10^6$	$2.9 \cdot 10^7$
	0.4	1.0	4.2	17.3	72.6	297.1
Prop. + PELT	$1.8 \cdot 10^4$	$4.3 \cdot 10^4$	$1.7 \cdot 10^5$	$6.8 \cdot 10^5$	$2.7 \cdot 10^6$	$9.5 \cdot 10^6$
	0.3	0.6	2.2	8.7	34.5	120.9
<b>Increasing number of discontinuities</b>						
Baseline	$3.9 \cdot 10^5$	$8.0 \cdot 10^5$	$7.4 \cdot 10^5$	$1.4 \cdot 10^6$	$2.9 \cdot 10^6$	$5.7 \cdot 10^6$
	28.6	57.8	81.8	158.9	323.6	652.3
Prop. + FPVI	$2.1 \cdot 10^4$	$9.1 \cdot 10^4$	$1.6 \cdot 10^5$	$2.4 \cdot 10^5$	$5.0 \cdot 10^5$	$1.3 \cdot 10^6$
	0.2	1.0	1.7	2.5	5.3	13.7
Prop. + PELT	$1.2 \cdot 10^4$	$2.4 \cdot 10^4$	$3.5 \cdot 10^4$	$6.9 \cdot 10^4$	$1.4 \cdot 10^5$	$2.8 \cdot 10^5$
	0.2	0.3	0.5	1.0	2.0	3.9

Upper rows: frequency of data point “visits”, Lower rows: runtime in seconds

# Hyperparameter selection

Use standard K-fold cross validation:

- partition data into  $K$  folds of approximately equal size
- select parameters that have a minimum K-fold cross validation score given by

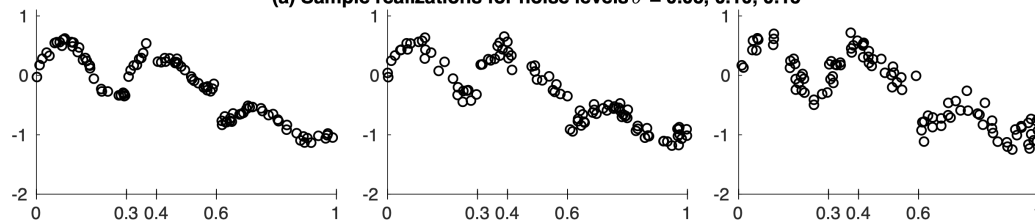
$$\text{CV}(p, \gamma) = \frac{1}{N} \sum_{k=1}^K \sum_{i \in \text{Fold}_k} ((\hat{f}_{p,\gamma}^{-k}(x_i) - y_i) / \delta_i)^2.$$

where  $\hat{f}_{p,\gamma}^{-k}$  is the CSSD results excluding the data in fold  $k$ .

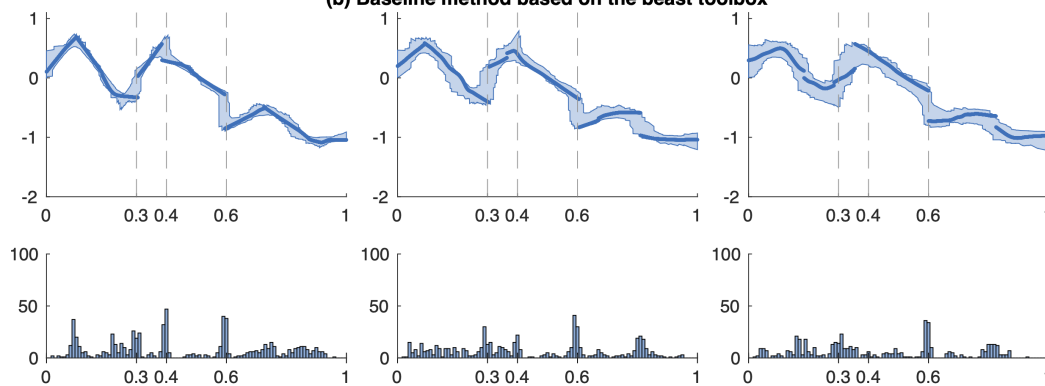
- search for minimizer using simulated annealing
- Computationally feasible because of fast algorithm for computing the CSSD result

$$\hat{f}_{p,\gamma}^{-k}$$

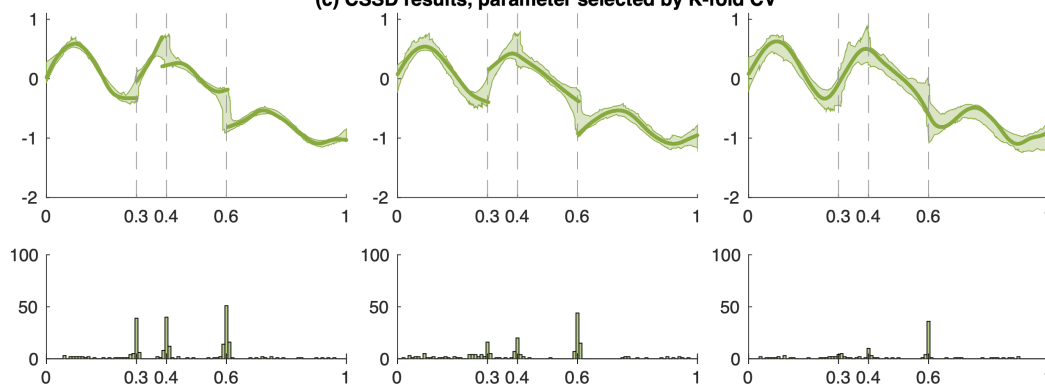
(a) Sample realizations for noise levels  $\sigma = 0.05, 0.10, 0.15$



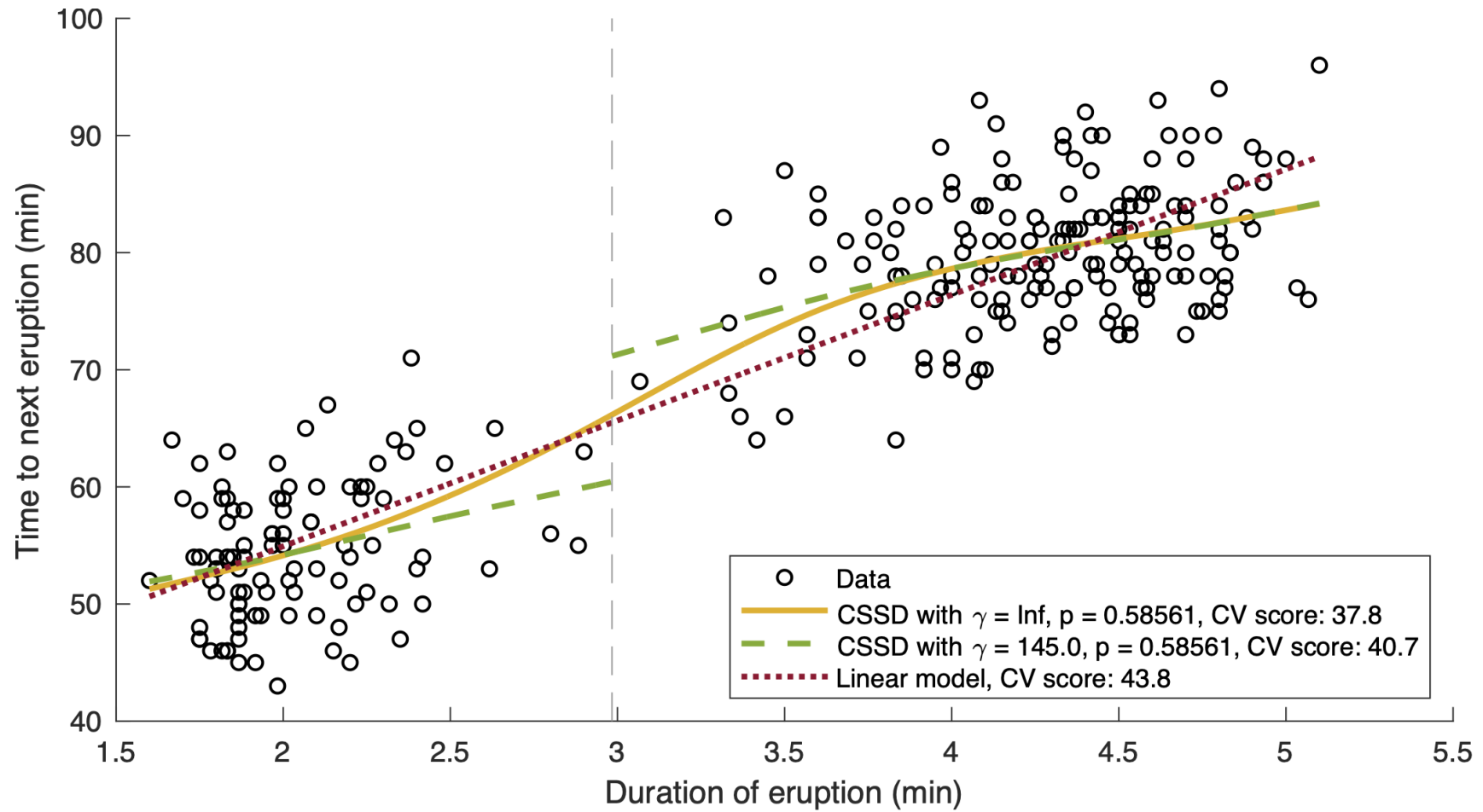
(b) Baseline method based on the beast toolbox



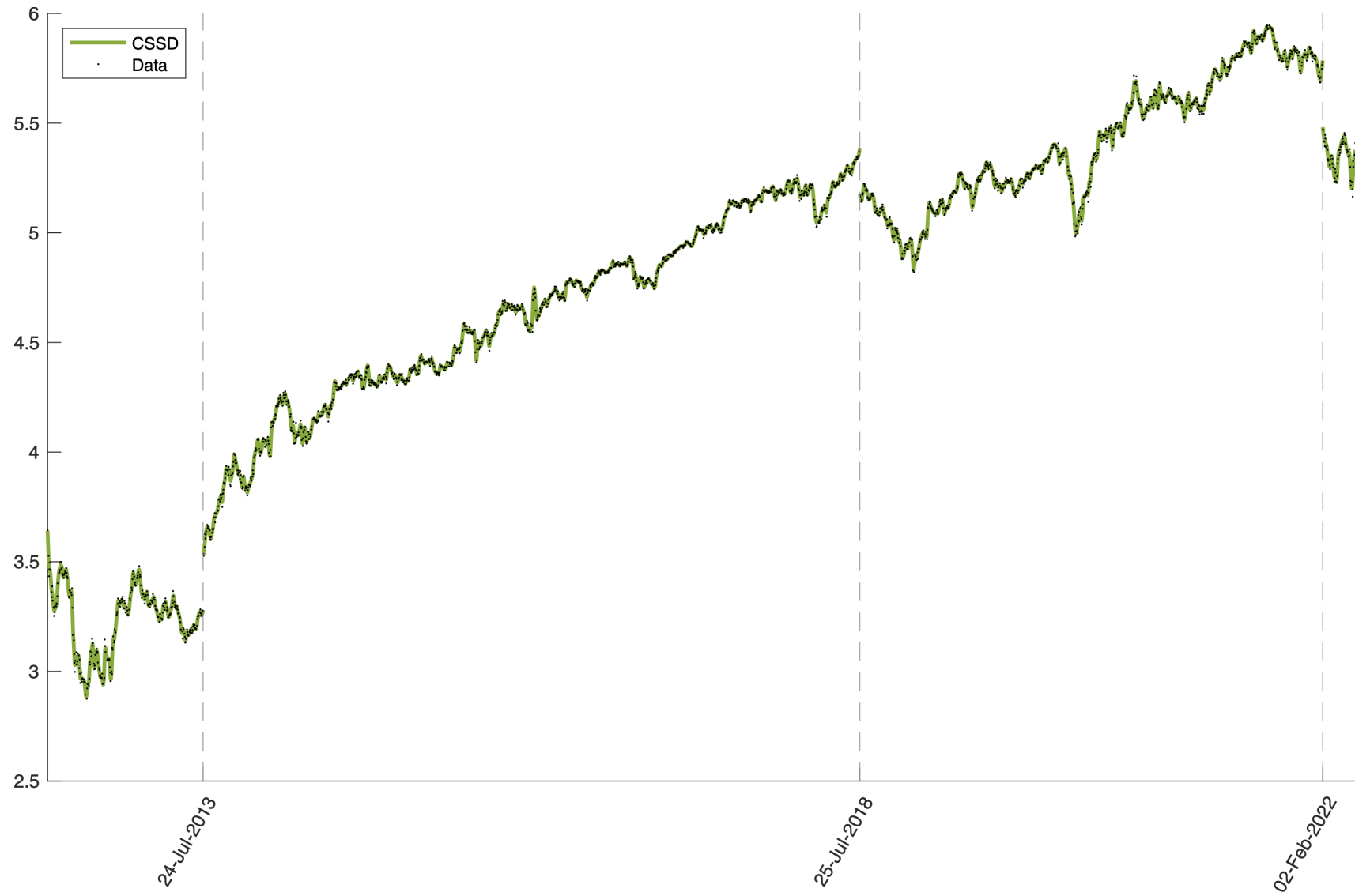
(c) CSSD results, parameter selected by K-fold CV



*Results on simulated data*



*Results on Old Faithful geyser data*



*Results on closing prices of the Meta stock from May 18, 2012, to May 19, 2022 (log scale)*

# The 2D Case: Potts model for unsupervised image segmentation

Potts model or piecewise constant Mumford-Shah model in 2D

$$\arg \min_u \gamma \|\nabla u\|_0 + \|Au - f\|_2^2$$

- $\|\nabla u\|_0$  total length of the jump set of  $u$ , weighted by  $\gamma > 0$
- $f$  aquired data,  $A$  linear operator



*Case  $A = id$ : Original  $f$ , solution  $u^*$ , jump set*



## Selected prior work

- For  $A = id$ 
  - first used for image segmentation by Geman and Geman (1984)
  - continuous variant studied by Mumford and Shah (1985,1989)
  - Algorithms: Blake/Zisserman, Boykov et al., Chan/Vese, Chambolle et al., Pock et al., Nikolova et al., Hirschmüller, Lellmann/Schnörr, Strekalovskiy et al., Xu et al.,...
  - NP-hard problem (Veksler 1999)  $\rightsquigarrow$  approximative strategies
- For  $A \neq id$ 
  - Existence of minimizers (Rondi/Santosa '01; Ramlau/Ring '07,'10; Fornasier/Ward '10; Jiang/Maaß/Page '14)
  - Regularizing properties (Ramlau/Ring '07,'10; Jiang/Maaß/Page '14)
  - Ambrosio-Tortorelli approximation (Rondi/Santosa '01; Bar/Sochen/Kiryati '04)
  - Level set active contours (Kim et al. '02; Ramlau/Ring '07; Klann/Ramlau/Ring '11)
  - Graduated non-convexity (Nikolova et al.~'08,'10)
  - Iterative thresholding (Fornasier/Ward '10)

# Discretization of the jump penalty

## Discretization

$$u^* = \arg \min_{u \in \mathbb{R}^{m \times n}} \gamma \sum_{s=1}^S \omega_s \|\nabla_{p_s} u\|_0 + \|u - f\|_2^2$$

with a finite difference system  $\mathcal{N} = p_1, \dots, p_S \subset \mathbb{Z}^2 \setminus \mathbf{0}$ , weights  $\omega_1, \dots, \omega_S > 0$ , and  $\|\nabla_{p_s} u\|_0 = |\{(i, j) : u_{(i,j)+p_s} \neq u_{(i,j)}\}|$ .

**Common finite difference systems** (Blake/Zisserman '87; Chambolle '99)

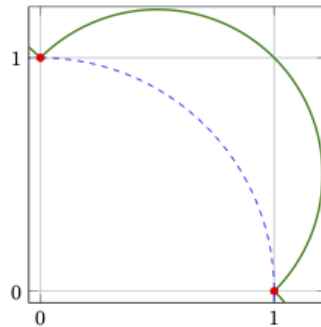
$$\mathcal{N}_0 = \{(1, 0), (0, 1)\},$$

$$\mathcal{N}_1 = \{(1, 0), (0, 1), (1, 1), (1, -1)\},$$

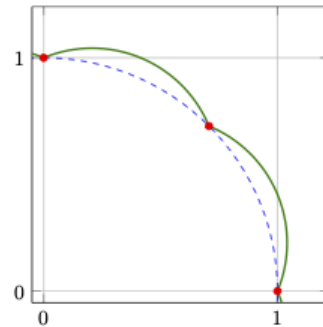
$$\mathcal{N}_2 = \{(1, 0), (0, 1), (1, 1), (1, -1), (-2, -1), (-2, 1), (2, 1), (2, -1)\}.$$

Proposed design criterion (S./Weinmann/Friel/Unser, Inv. Prob. 2015)

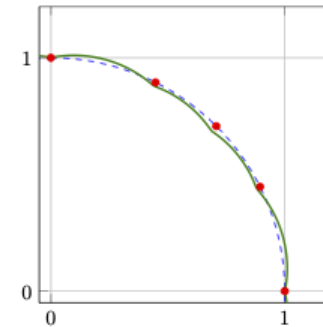
$$\sum_{s=1}^S \omega_s |\langle p, p_s \rangle| \stackrel{!}{=} \|p\|_2 \quad \text{for all } p \in \mathcal{N}.$$



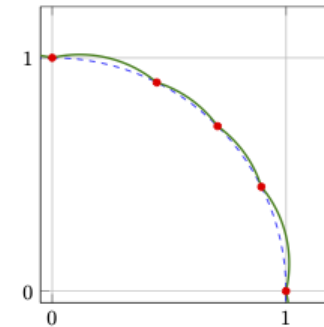
Anisotropic  
( $E \approx 1.41$ ).



With diagonals  
( $E \approx 1.08$ )

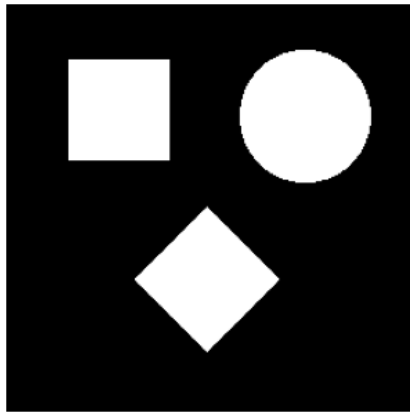


“Knight moves”, weights of  
Chambolle '99  
( $E \approx 1.05$ )

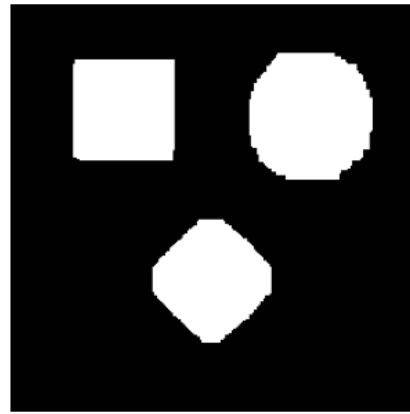


“Knight moves”,  
our weights  
( $E \approx 1.03$ )

$E$  = Ratio of longest and shortest vector on the green line  
(measure of isotropy used in Chambolle 1999,  $E = 1$  is optimal)



Original



Reconstruction using  $\mathcal{N}_0$



Reconstruction using  $\mathcal{N}_1$



Reconstruction using  $\mathcal{N}_2$

(Setup: Reconstruction from Radon data, 4 angular projections, noise level 0.01.)

~> “Knight-move” neighborhood system  $\mathcal{N}_2$  exhibits almost no anisotropy effects

## ADMM-Algorithm based on directional splitting

$$\begin{aligned} & \underset{u_1, \dots, u_S, v}{\text{minimize}} && \gamma \sum_{s=1}^S \omega_s \|\nabla_{p_s} u_s\|_0 + \|Av - f\|_2^2 \\ & \text{subject to} && u_r - u_t = 0, \text{ for all } 1 \leq r < t \leq S, \\ & && v - u_s = 0, \text{ for all } 1 \leq s \leq S. \end{aligned}$$

Augmented Lagrangian

$$\begin{aligned} \mathcal{L}(\{u_s\}_{s=1}^S, v, \{\lambda_s\}_{s=1}^S, \{\rho_{r,t}\}_{1 \leq r < t \leq S}) = & \\ & \sum_{s=1}^S \omega_s \|\nabla_{p_s} u_s\|_0 + \frac{\mu}{2} \|v - u_s + \frac{\lambda_s}{\mu}\|_2^2 - \frac{1}{2\mu} \|\lambda_s\|_2^2 \\ & + \sum_{1 \leq r < t \leq S} \frac{\nu}{2} \|u_r - u_t + \frac{\rho_{r,t}}{\nu}\|_2^2 - \frac{1}{2\nu} \|\rho_{r,t}\|_2^2 + \|Av - f\|_2^2, \end{aligned}$$

- $\{\lambda_s\}_{1 \leq s \leq S}$  and  $\rho_{r,t}_{1 \leq r < t \leq S}$  are Lagrange multipliers,  $\mu, \nu > 0$  regulate coupling

## ADMM-type algorithm

$$\left\{ \begin{array}{l}
 u_1^{k+1} \in \arg \min_{u_1} \frac{2\gamma\omega_1}{\mu_k + \nu_k(S-1)} \|\nabla_{p_1} u_1\|_0 + \|u_1 - w_1^k\|_2^2, \\
 \vdots \\
 u_S^{k+1} \in \arg \min_{u_S} \frac{2\gamma\omega_S}{\mu_k + \nu_k(S-1)} \|\nabla_{p_S} u_S\|_0 + \|u_S - w_S^k\|_2^2, \\
 v^{k+1} = \arg \min_v \|Av - f\|_2^2 + \frac{\mu_k S}{2} \|v - z^k\|_2^2, \\
 \lambda_s^{k+1} = \lambda_s^k + \mu_k(v^{k+1} - u_s^{k+1}), \quad \text{for all } 1 \leq s \leq S, \\
 \rho_{r,t}^{k+1} = \rho_{r,t}^k + \nu_k(u_r^{k+1} - u_t^{k+1}), \quad \text{for all } 1 \leq r < t \leq S
 \end{array} \right.$$

# Convergence of the ADMM-type algorithm

Theorem (S./Weinmann/Friel/Unser, Inv Prob 2015)

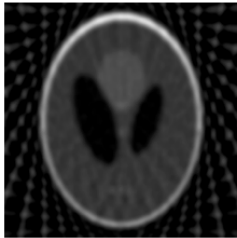
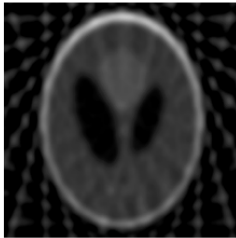
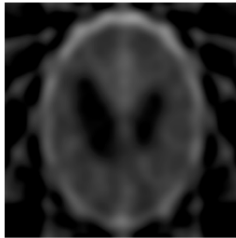

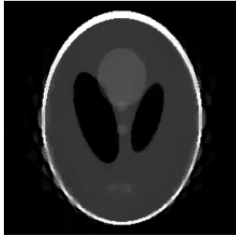
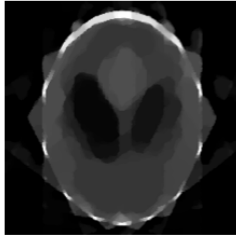



Let the sequence  $(\mu_k)_{k \in \mathbb{N}}$  be increasing and satisfy  $\sum_k \mu_k^{-1/2} < \infty$ . Then, the proposed iteration converges in the sense that

$$(u_1^k, \dots, u_S^k, v^k) \rightarrow (u_1^*, \dots, u_S^*, v^*) \quad \text{with} \quad u_1^* = \dots = u_S^* = v^*,$$

$$\text{and} \quad \frac{\lambda_s^k}{\mu_k} \rightarrow 0 \quad \text{for all} \quad s \in \{1, \dots, S\}.$$

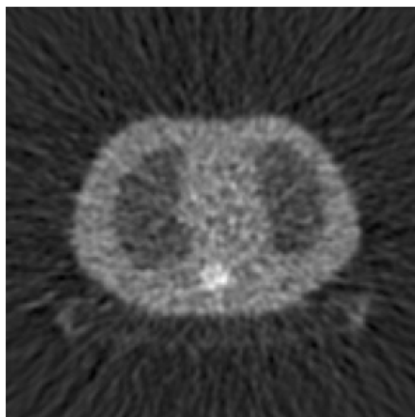
- Best quality for slow progression, e.g.,  $\mu_k = k^{2.01} \cdot 10^{-6}$
- Earlier convergence for fast progression, e.g.,  $\mu_k = 1.05^k \cdot 10^{-6}$
- For convergence guarantees to local minimum can use a surrogate-type algorithm [Kiefer/S./Weinmann, FoCM 2021]

# Joint image reconstruction and segmentation from sparse Radon data

	17 Angles	12 Angles	7 Angles
FBP (tuned PSNR) w.r.t.			
Total variation regularization			
Proposed method			



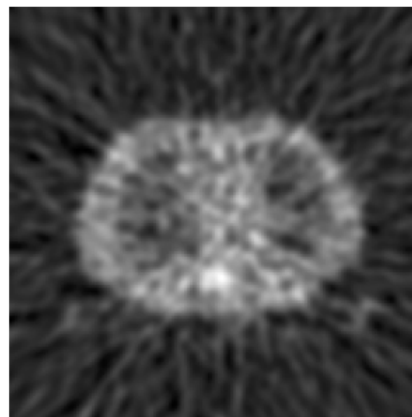
# Joint image reconstruction and segmentation from sparse PET data



FBP using 192 projections.



Proposed method using 192 projections.



FBP using 24 projections.



Proposed method using 24 projections.

*PET data of a physical thorax phantom, courtesy of Jeffrey Fessler*

# Higher order methods

Piecewise affine linear Potts model

$$\arg \min_{u, \mathcal{P}} \sum_{P \in \mathcal{P}} \left\{ \frac{\gamma}{2} \text{length}(\partial P) + \int_P |u(x) - f(x)|^2 dx \right\},$$

subject to  $u|_P$  is affine linear for all  $P \in \mathcal{P}$ .



(a) Input



(b) Pcw. constant model



(c) Pcw. affine model

# Computational approaches

- Iterated graph cuts (Yang/Li, 2015)
- Active contours (Wang et al., 2013)
- ADMM splitting for piecewise affine Potts model applied to optical flow/motion field images (Fortun et al., IEEE TIP 2018)



Overlay of the input images



Ground truth



Our method



*TGV-Census*



*NLTV-Census*



*DataFlow*

“Potts-ADMM” gives good resulting images also for the piecewise linear model ...



... but corresponding edges sometimes broken

## Proposed solution: Taylor jet coupling approach

First order jet  $\mathcal{J}u$  of a function  $u$  at the point  $x \in \Omega$

$$\mathcal{J}u(x)(z) := T_x u(z) = u(x) + \frac{\partial u(x)}{\partial x_1} (z_1 - x_1) + \frac{\partial u(x)}{\partial x_2} (z_2 - x_2)$$

Pcw. affine constraint becomes piecewise constant in terms of the jet

$$\arg \min_{u, P_c} \sum_{P \in P_c} \left\{ \frac{\gamma}{2} \text{length}(\partial P) + \int_P (u(x) - f(x))^2 dx \right\},$$

subject to  $\mathcal{J}u|_P$  is constant for all  $P \in \mathcal{P}$ .

Formulate affine linear model as piecewise constant jet problem (Kiefer/S./Weinmann, IEEE TIP 2020)

$$J^* = \arg \min_{J \in PC(\Omega; \Pi_1)} \gamma \|\nabla J\|_0 + \int_{\Omega} (J(x)(x) - f(x))^2 dx.$$

## Jet based splitting

$$\arg \min_{J^1, \dots, J^S} \sum_{s=1}^S \left\{ \gamma \omega_s \|\nabla_{d^s} J^s\|_0 + \frac{1}{S} \sum_{x \in \Omega'} (J^s(x)(x) - f(x))^2 \right\}$$

subject to  $J^s = J^t$  for all  $1 \leq s < t \leq S$ .

Applying ADMM gives

$$(J^s)^{j+1} = \arg \min_J \frac{2\omega_s \gamma}{(S-1)\nu_j} \|\nabla_{d^s} J\|_0 + \frac{2+\mu_j S}{\nu_j S} \|u - (w^s)^j\|^2$$
$$+ \|a - (y^s)^j\|^2 + \|b - (z^s)^j\|^2 \quad \forall s = 1, \dots, S,$$

$$(\lambda^{s,t})^{j+1} = (\lambda^{s,t})^j + \mu_j ((u^s)^{j+1} - (u^t)^{j+1}) \quad \forall s < t,$$

$$(\tau^{s,t})^{j+1} = (\tau^{s,t})^j + \nu_j ((a^s)^{j+1} - (a^t)^{j+1}) \quad \forall s < t,$$

$$(\rho^{s,t})^{j+1} = (\rho^{s,t})^j + \nu_j ((b^s)^{j+1} - (b^t)^{j+1}) \quad \forall s < t$$

↪ **linewise segmented jet problems** which can be solved exactly using dynamic programming. (Kiefer/S./Weinmann, IEEE TIP 2020)

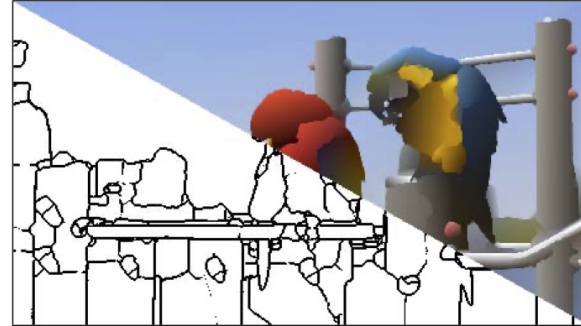
# Results of the jet splitting approach



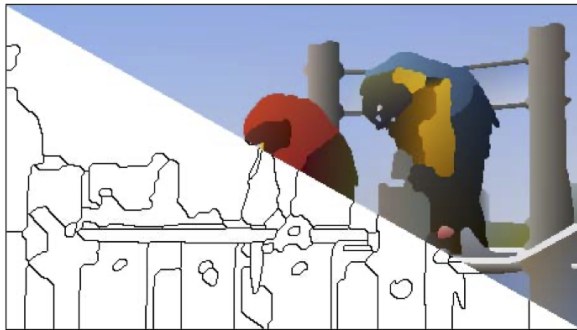
(a) Input



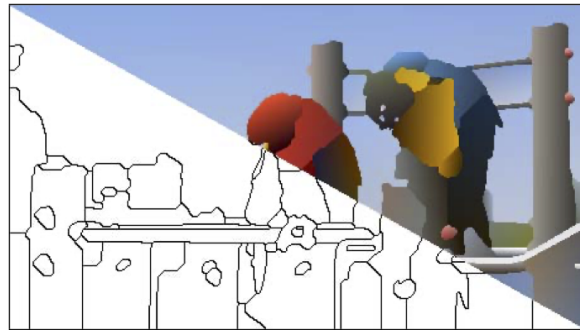
(b) GNC



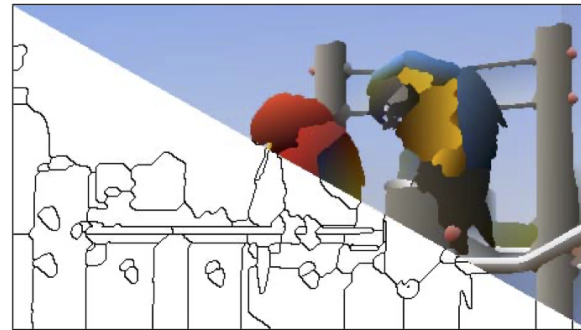
(c) FPAME



(d) GC (64 labels)



(e) GC (256 labels)



(f) Proposed

Edge sets are now closed as desired

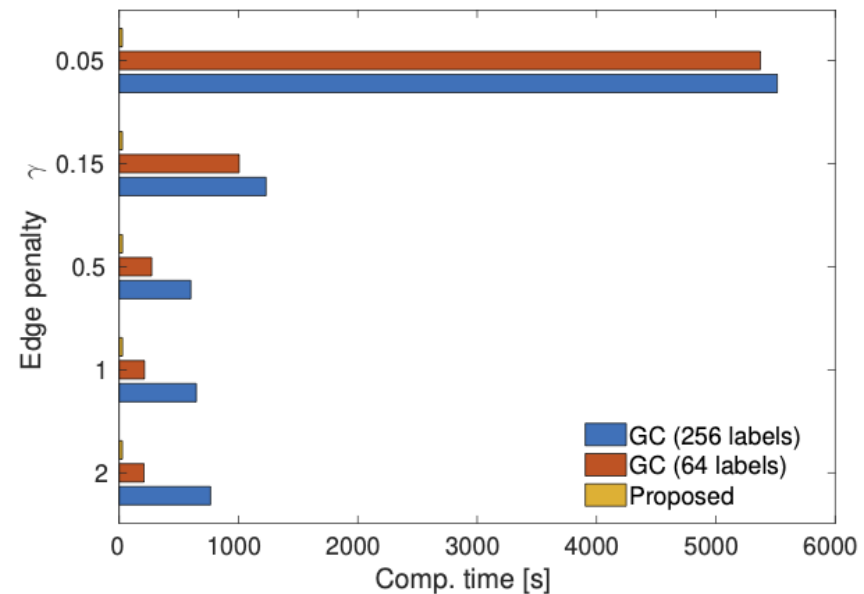
# Quantitative comparison with iterated graph cuts

- Comparison to iterative repartitioning with graph cut algorithm  $\alpha$ -expansion (Boykov et al. 2001, 2004) on Berkley dataset
- proposed method is more effective and considerably faster

*Functional values*

$\gamma$	GC (64 labels)	GC (256 labels)	Proposed
0.05	1802.2	1802.1	<b>1772.3</b>
0.15	3180.3	3178.2	<b>3146.0</b>
0.50	5484.8	5480.7	<b>5418.2</b>
1.00	7324.7	7317.5	<b>7202.7</b>
2.00	9551.5	9526.4	<b>9356.7</b>

*Computation times*



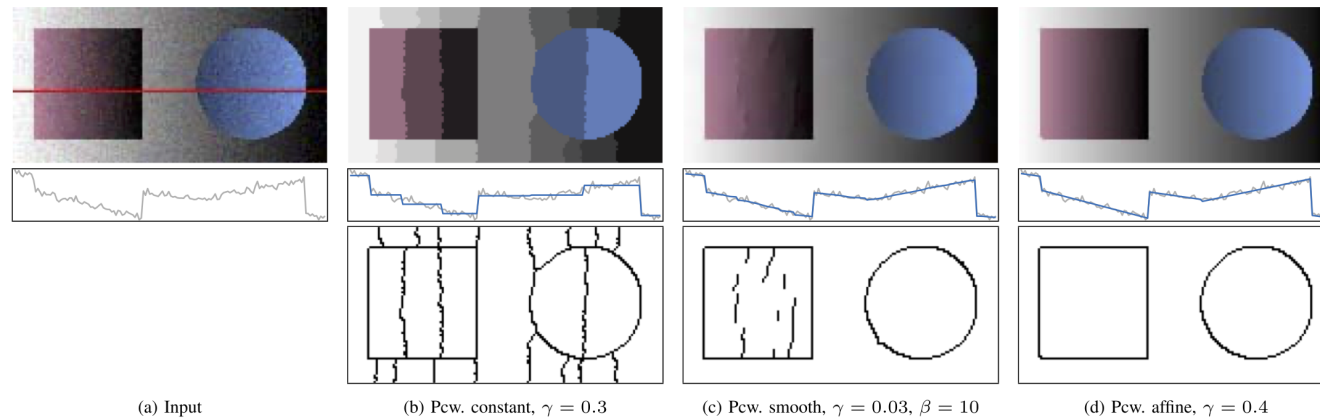


# Piecewise smooth models in 2D

Mumford-Shah model (weak membrane model)

$$\min_{u, \Gamma} \|u - f\|^2 + \beta^2 \int_{\Omega \setminus \Gamma} \|\nabla u(x)\|^2 dx + \gamma \cdot \text{length}(\Gamma)$$

First order Mumford-Shah model exhibits the so-called “gradient-limit effect”, i.e., it produces extra edges at steep slopes (Blake/Zisserman 1987)



## Second order methods

Second order Mumford-Shah model (weak plate model without creases)

$$\min_{u, \Gamma} \|u - f\|^2 + \beta^2 \int_{\Omega \setminus \Gamma} \|\nabla^2 u(x)\|_F^2 dx + \gamma \cdot \text{length}(\Gamma)$$

Algorithmic approaches

- Graduated non-convexity (Blake/Zisserman 1987)
- Ambrosio-Tortorelli type approximation (Zanetti/Bruzzone 2017)
- Taylor-Jet approach (Kiefer/S./Weinmann 2020)



(a) Noisy data



(b) Clean image (groundtruth)



(c) First order Mumford-Shah model



(d) Second order Mumford-Shah model  
(using the prop. meth.)



(e) Detail of (a)



(f) Detail of (b)

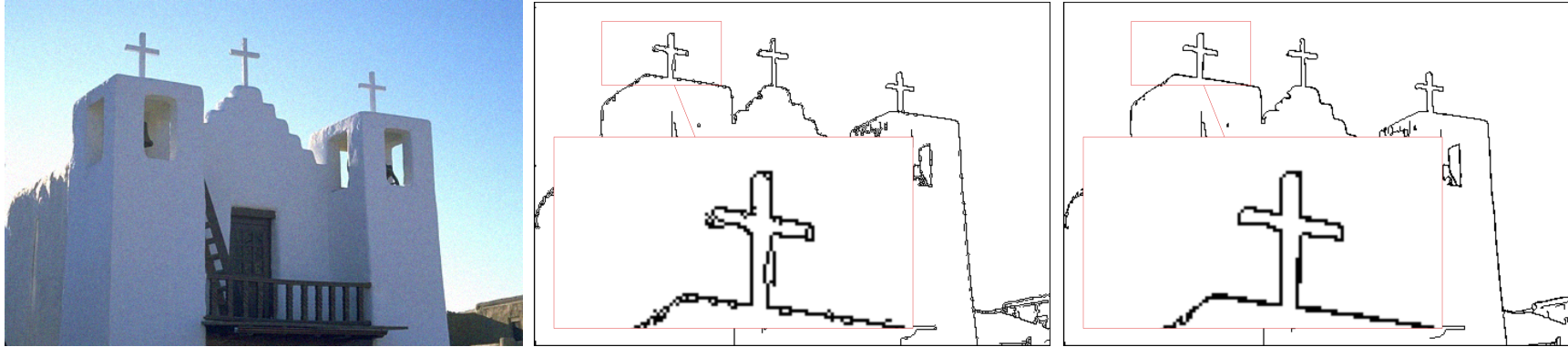


(g) Detail of (c)



(h) Detail of (d)

# Effects of the jet splitting on the resulting edge set



Edge set with pixel coupling and with Jet coupling

# Qualitative comparison



(a) Input



(b) GNC



(c) Edges of (b)



(d) Details of (b) and (c)



(e) Ambrosio-Tortorelli



(f) Edges of (e)



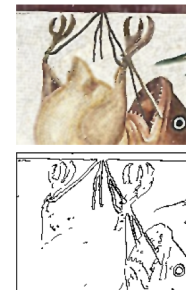
(g) Details of (e) and (f)



(h) Proposed algorithm



(i) Edges of (h)



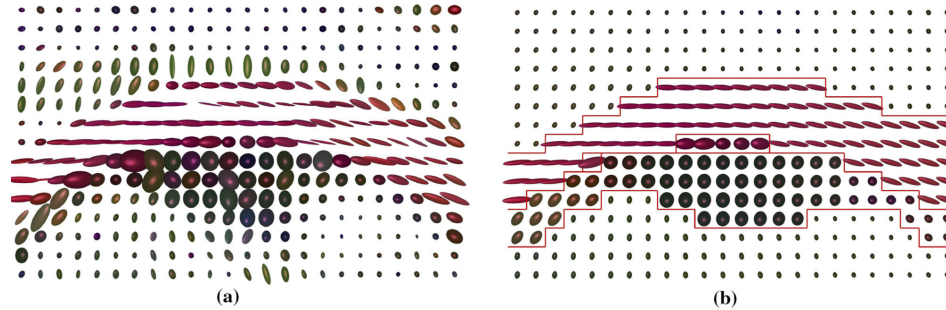
(j) Details of (h) and (i)

# Quantitative comparison

$\gamma$	$\beta$	GNC	AT	Prop.
0.025	0.5	1460.1	1228.6	<b>1064.2</b>
	1	2732.4	1944.2	<b>1474.0</b>
	2	5409.6	2837.3	<b>1783.0</b>
	3	8729.8	3519.0	<b>1918.2</b>
	5	17148.0	4787.3	<b>2049.3</b>
	8	32323.7	6874.9	<b>2141.1</b>
0.035	0.5	1493.6	1312.9	<b>1138.2</b>
	1	2842.9	2136.9	<b>1624.0</b>
	2	5815.9	3176.4	<b>2014.3</b>
	3	9536.4	3955.2	<b>2193.2</b>
	5	17863.2	5288.9	<b>2371.2</b>
	8	33581.0	7280.7	<b>2498.8</b>
0.05	0.5	1500.0	1376.5	<b>1200.4</b>
	1	2878.4	2301.4	<b>1760.9</b>
	2	5877.3	3518.4	<b>2252.1</b>
	3	9630.9	4433.9	<b>2486.1</b>
	5	19677.9	5959.5	<b>2728.3</b>
	8	34300.2	7950.1	<b>2904.8</b>
0.1	0.5	1480.5	1390.9	<b>1262.2</b>
	1	2767.1	2377.8	<b>1928.3</b>
	2	5790.6	3858.3	<b>2595.0</b>
	3	10317.9	5111.4	<b>2953.7</b>
	5	21490.8	7229.5	<b>3354.2</b>
	8	38044.1	9857.4	<b>3672.6</b>

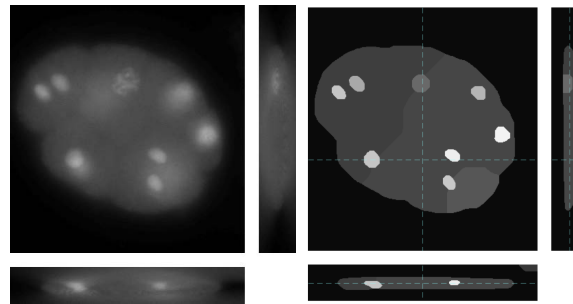
# Extensions

- Mumford-Shah and Potts regularization for manifold-valued data (Baust et al. 2016, Weinmann et al. 2016)



*Corpus callosum of a human brain from the Camino project and Mumford-Shah regularization*

- Fast algorithm for Potts model on 3D data with GPU parallelization (S. et al. 2017)



*Microscopy image of a C. elegans embryo and result of joint 3D deconvolution and segmentation*



# References

## Own papers

- M. Storath, A. Weinmann. Smoothing splines for discontinuous signals. *Journal of Computational and Graphical Statistics*. 2023
- L. Kiefer, M. Storath, A. Weinmann. An algorithm for second order Mumford-Shah models based on a Taylor jet formulation. *SIAM Journal on Imaging Sciences*, 2020
- L. Kiefer, M. Storath, A. Weinmann. PALMS Image Partitioning - A New Parallel Algorithm for the Piecewise Affine-Linear Mumford-Shah Model. *IPOL*, 2020
- L. Kiefer, M. Storath, A. Weinmann. Iterative Potts minimization for the recovery of signals with discontinuities from indirect measurements – the multivariate case. *Foundations of Computational Mathematics*, 2020
- L. Kiefer, M. Storath, A. Weinmann. An efficient algorithm for the piecewise affine-linear Mumford-Shah model based on a Taylor jet splitting. *IEEE T on Image Processing*, 2019
- M. Storath, L. Kiefer, A. Weinmann. Smoothing for signals with discontinuities using higher order Mumford-Shah models. *Numerische Mathematik*, 2019
- D. Fortun, M. Storath, D. Rickert, A. Weinmann, M. Unser. Fast piecewise-affine motion estimation without segmentation. *IEEE T on Image Processing*, 27(11):5612–5624, 2018
- M. Storath, A. Weinmann. Fast partitioning of vector-valued images. *SIAM Journal on Imaging Sciences*, 2014.



## Software

- [https://github.com/lu-kie/PALMS\\_ImagePartitioning](https://github.com/lu-kie/PALMS_ImagePartitioning) (Piecewise affine linear model, C++/Matlab)
- <https://ipolcore.ipol.im/demo/clientApp/demo.html?id=295> (Online Demo for PALMS)
- <https://github.com/mstorath/CSSD> (CSSD in Matlab)
- [https://github.com/lu-kie/HOMS\\_SignalProcessing](https://github.com/lu-kie/HOMS_SignalProcessing) (Higher order Mumford-Shah in C++)
- <https://github.com/mstorath/Pottslab> (Potts and Mumford-Shah in Java/Matlab)
- <https://icy.bioimageanalysis.org/plugin/potts-segmentation/> (Icy Plugin)

## Related Work (non-exhaustive selection)

- D. Mumford, J. Shah. Boundary detection by minimizing functionals. IEEE Conference on Computer Vision and Pattern Recognition, 1985.
- A. Blake, A. Zisserman. Visual Reconstruction. MIT Press Cambridge, 1987.
- D. Mumford, J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. Comm on Pure and Applied Mathematics, 1989.
- L. Ambrosio, V. Tortorelli. Approximation of functional depending on jumps by elliptic functional via  $\Gamma$ -convergence. Comm on Pure and Applied Mathematics, 1990.
- A. Chambolle. Finite-differences discretizations of the Mumford-Shah functional. ESAIM: Mathematical Modelling and Numerical Analysis, 1999.
- T. Chan, L. Vese. A level set algorithm for minimizing the Mumford-Shah functional in image processing. IEEE Workshop on Variational and Level Set Methods in Computer Vision, 2001.
- G. Winkler, V. Liebscher. Smoothers for discontinuous signals. Journal of Nonparametric Statistics, 2002.
- F. Friedrich, A. Kempe, V. Liebscher, and G. Winkler. Complexity penalized M-estimation: fast computation. J Computational and Graphical Statistics, 2014.
- M. Zanetti, L. Bruzzone. Piecewise linear approximation of vector-valued images and curves via second-order variational model. IEEE T Image Processing, 2017.
- M. Foare, N. Pustelnik, and L. Condat. Semi-linearized proximal alternating minimization for a discrete Mumford-Shah model. IEEE T on Image Processing, 2019.