Curvilinear Structure Modeling and Its Applications in Computer Vision

Seong-Gyun JEONG

seong-gyun.jeong@inria.fr

In collaboration with Dr. Josiane ZERUBIA and Dr. Yuliya TARABALKA

AYIN, INRIA Sophia Antipolis Méditerranée, France

Publications available on https://team.inria.fr/ayin/publications-hal/

Outline

1. Introduction

- Motivations
- Previous work
- Overview

2. Stochastic model

- Curvilinear features
- Stochastic model
- Reversible jump Markov chain Monte Carlo
- Integration of line hypotheses

3. Structured ranking learning

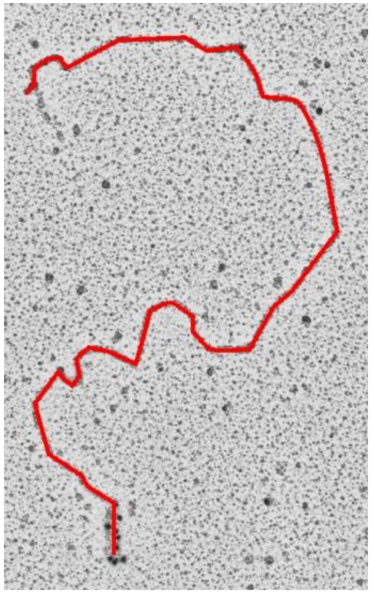
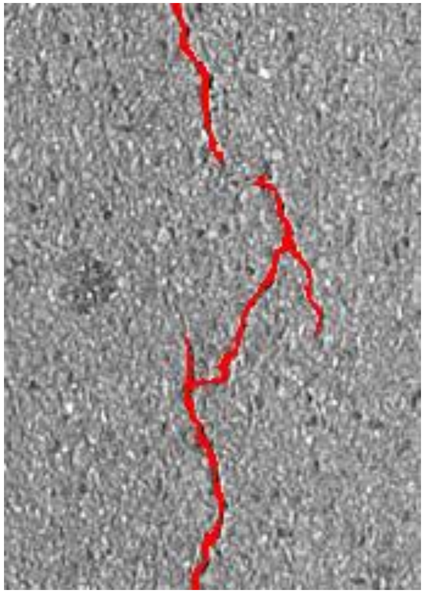
- Orientation-aware curvilinear features
- Structured ranking learning for curvilinear structure reconstruction
- Progressive curvilinear path reconstruction

Introduction

Motivations

- Appears in various types of natural images

  e.g., retina, DNA, road network, cracks, facial wrinkles, ...
- Shows complex geometry
- Low contrast, surrounded by the similar background textures

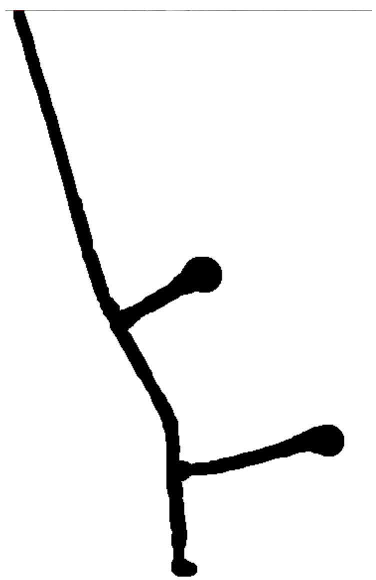
Previous work

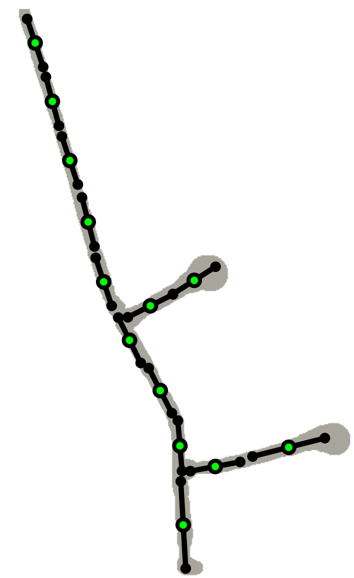
- Blood vessel segmentation [Frangi 98, Staal 04, Law 08, ...]
  - To enhance visibility and aid diagnoses of vascular diseases
- Bioimage analysis [Zhao 11, Wang 11, Peng 11, Turetken 13, ...]
  - To reconstruct physical structure of neural network
- Facial wrinkle detection [Batool 12, Jeong 14, ...]
  - To evaluate skin condition for beauty and dermatology
- Road network extraction [Lacoste 05, Hu 07, Valero 10, ...]
  - To extract geographical information from satellite images
- Defects in the asphalts [Iyer 05, Chambon 10, ...]
  - To analyze large surfaces safely
- **Issues**: Automatization + Generalization

  Reduce user-defined parameters

  Employ machine learning systems

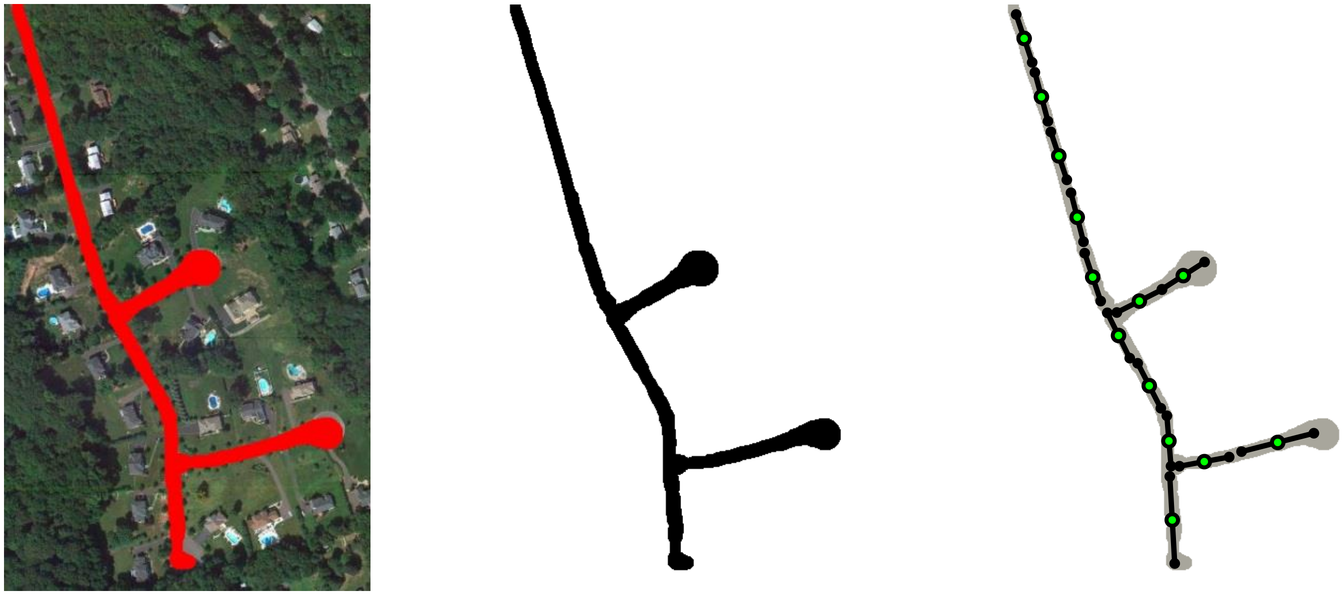
Overview





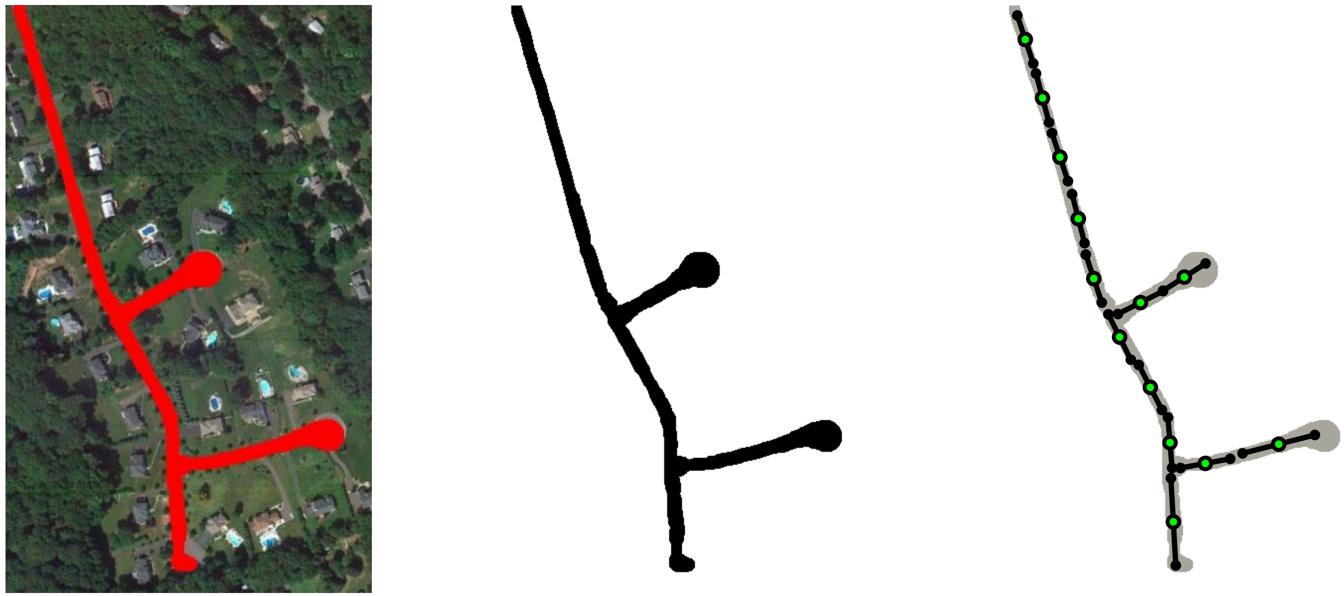
- **Goal**: **Unified framework** for curvilinear structure reconstruction
- **Assumption**: can be decomposed into multiple line segments
- Find an **optimal set of line segments** for curvilinear structure reconstruction

Overview



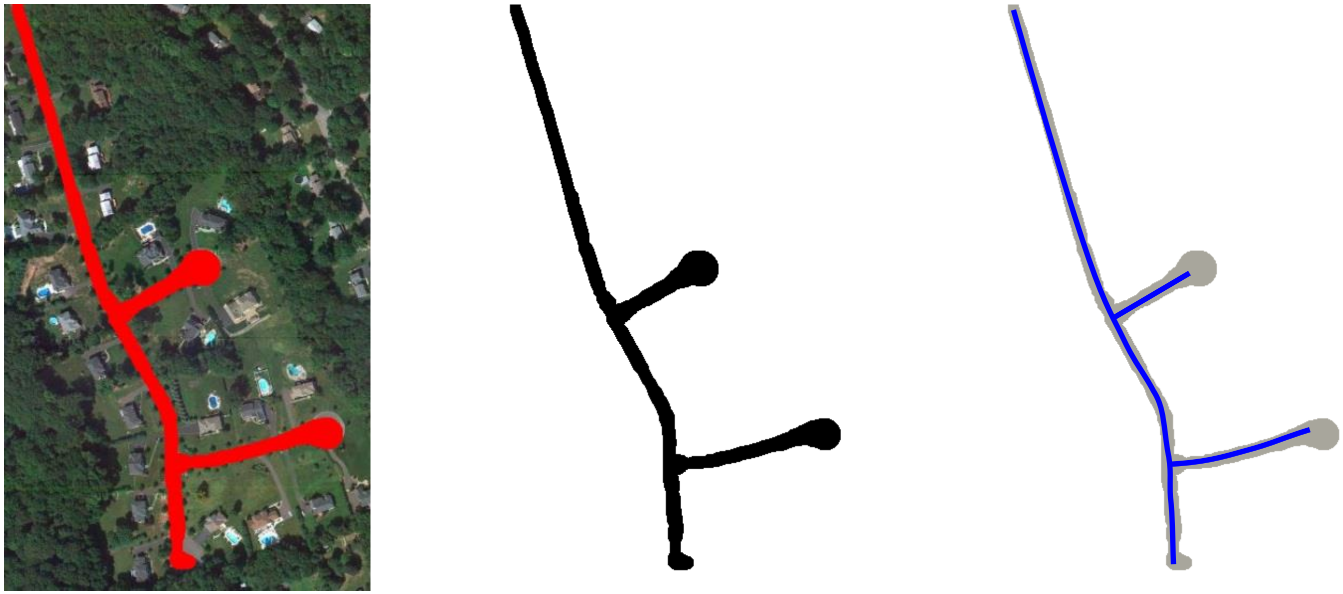
- **Stochastic model**
  - Maximize **a posterior probability** of line segments for given image
  - Data likelihood: *local curvilinear features*
  - Prior energy: *constrains local geometry of line segments*

Overview



- **Structured ranking learning**
  - Learn a function to evaluate correspondence between line segment and the underlying curvilinear structure
  - Orientation-aware curvilinear feature descriptor

Overview



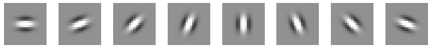
- **Progressive curvilinear path reconstruction**
  - Provide topological features and simplified curvilinear structure
  - Find the longest geodesic paths in the graph

Stochastic model

Curvilinear features

- Thin / Elongated / Symmetric / Locally oriented
- Show different intensity values compared to its surroundings

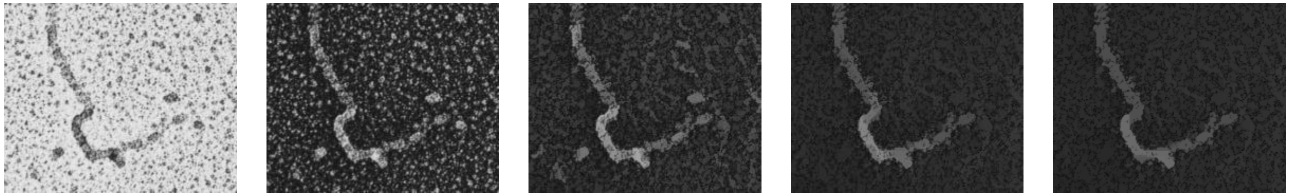
Image gradient

- Can measure local intensity variation
- 2nd derivative of Gaussian kernels 

Input 0^{o} 45^{o} 90^{o} 135^{o}

Morphological filtering [Talbot 07]

- Highlight structural similarity

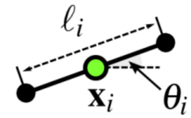


Input Invert L=10 L=50 L=100

Stochastic model

- Line segment as tuple of *pixel*, *length*, and *orientation*

$$s_i = (\mathbf{x}_i, \ell_i, \theta_i) \in \mathbb{R}^2 \times |L| \times |\Theta|$$

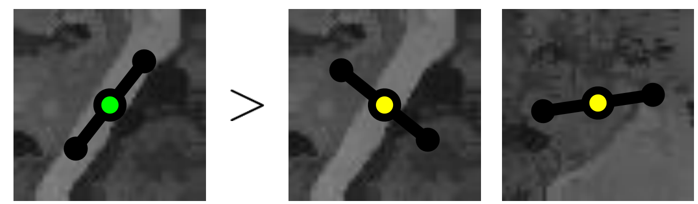


- Find a set of line segments that maximize a posterior probability

$$\hat{\mathbf{s}} = \operatorname*{argmax}_{\mathbf{s}\in\Psi} p(\mathbf{s}|I) = \operatorname*{argmax}_{\mathbf{s}\in\Psi} p(I|\mathbf{s})p(\mathbf{s})$$

Data likelihood

$$p(I|s_i) = \frac{1}{Z}\exp\left(-\sum_{\mathbf{x}_j\in s_i} E_{\text{data}}(\phi(\mathbf{x}_j))\right)$$



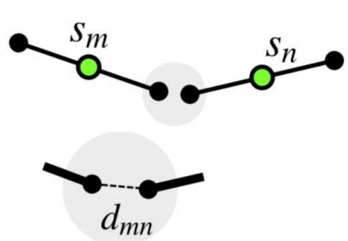
- To localize a line segment for a given image
- Curvilinear feature vector $\phi(\mathbf{x})$ evaluates a pixel whether it is on the curvilinear structure according to the image gradient and morphological filtering

Stochastic model

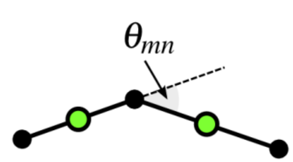
Prior energy

$$p(\mathbf{s}) = p(s_1, s_2, \ldots, s_N) = \prod_{s_m \sim s_n} p(s_m | s_n) = \sum_{s_m \sim s_n} E_{\text{prior}}(s_m, s_n)$$

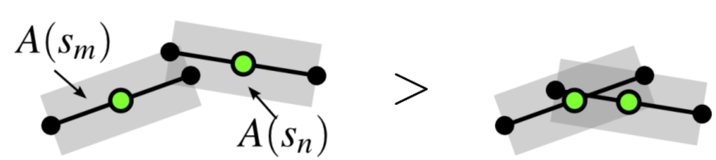
- Define ***local geometry*** of line segments *(line segments interact if they are close enough)*
- Smoothly connected line segments are desired
  - Connectivity: *end-to-end distance of line segments*
  - Curvature: *angle difference between adjacent line segments*
- Reject congestion of lines within local configuration
  - Measured by the proportion of pixels falling in the same areas



Connectivity



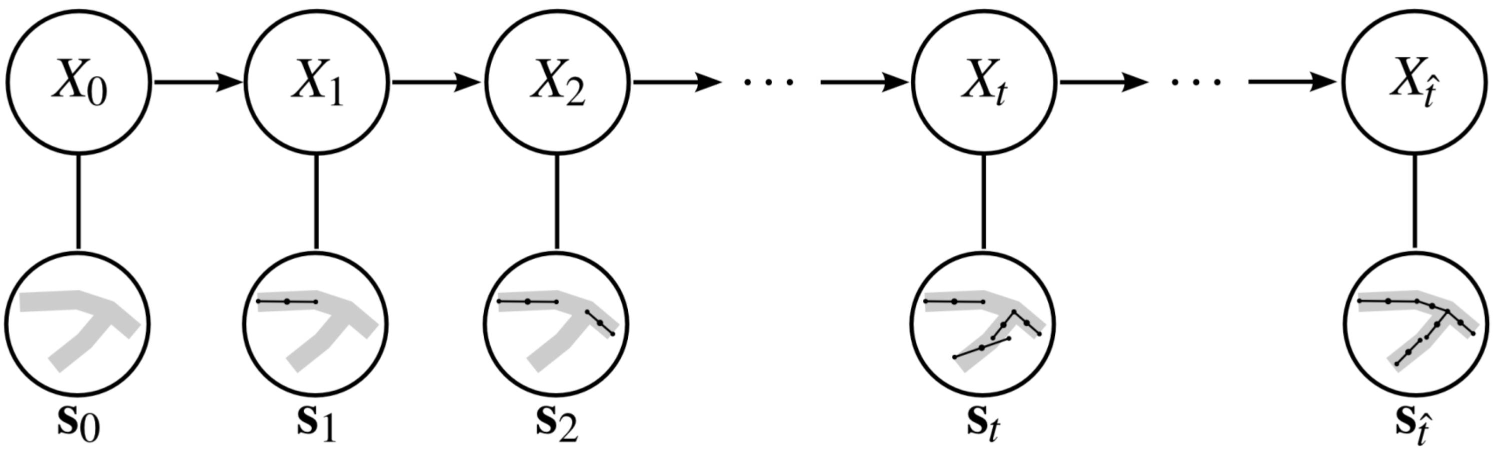
Curvature



Congestion

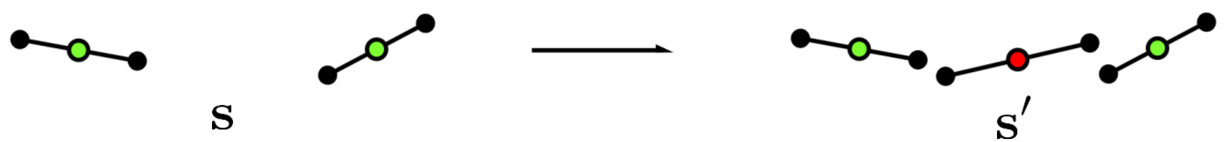
RJMCMC [Green 95]

- Reversible jump Markov chain Monte Carlo
- Each state of a discrete Markov chain $(X_t)_{t\in\mathbb{N}}$ corresponds to a random configuration on Ψ
- Markov chain eventually reaches an equilibrium state which maximizes the proposed density function

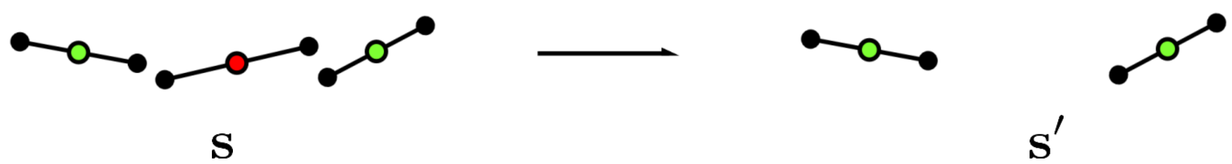


RJMCMC

- Simulate a discrete Markov chain over the configuration space via **sub-transition kernels** *i.e., Birth, Death, Affine transform*
- ***Birth kernel** proposes a new segment*
  - $s_i = (\mathbf{x}_i, \ell_i, \theta_i) \in \mathbb{R}^2 \times |L| \times |\Theta|$
  - Randomly select location, length, and orientation from sample space

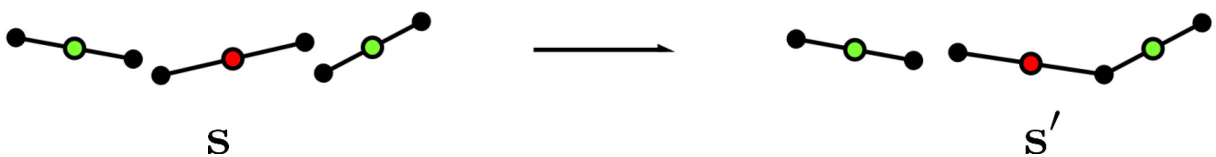


- ***Death kernel** removes a segment*
  - Randomly select an existing line segment from the current configuration

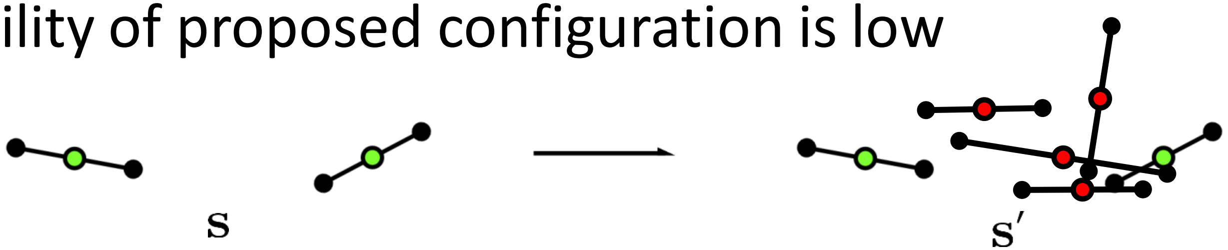


RJMCMC

- Simulate a discrete Markov chain over the configuration space via **sub-transition kernels** *i.e., Birth, Death, Affine transform*
- ***Affine transform*** *updates intrinsic variables of the segment*
  - Select a line segment and update its location, length, and orientation randomly
  - $s_i = (\mathbf{x}_i, \ell_i, \theta_i) \rightarrow s_i' = (\mathbf{x}_i \pm \triangle\mathbf{x}, \ell_i \pm \triangle\ell, \theta_i \pm \triangle\theta)$

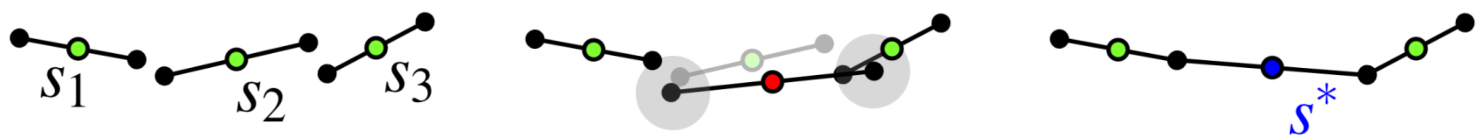


- Markov chain will ***remain at the current configuration***, if the probability of proposed configuration is low



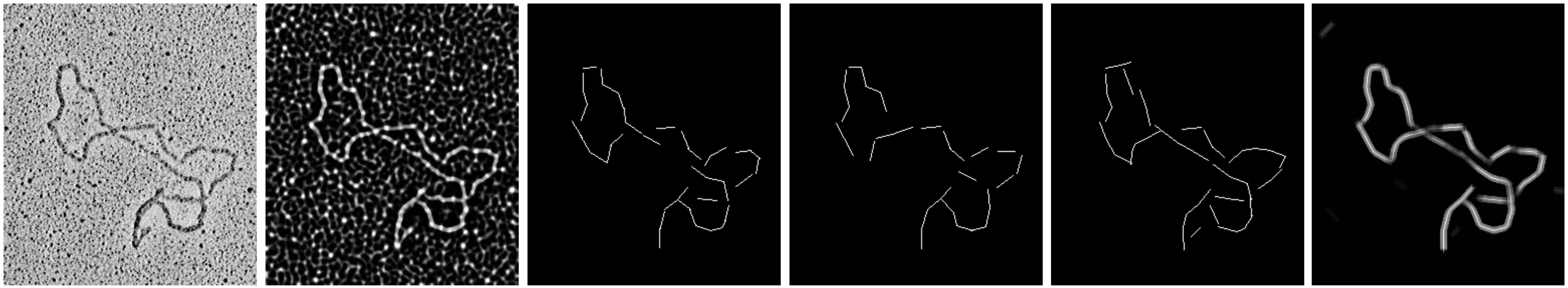
Delayed Rejection [Green 01]

- Gives a ***second chance*** to a rejected configuration ***by enforcing the connectivity***



1. Let $\mathbf{s}$={s_1, s_2, s_3} be the current configuration
2. Propose a new configuration via affine transform kernel
3. If s' is rejected, DR kernel searches for the nearest end points in the rest of the line segments
4. An alternative line segment s^* will enforce the connectivity

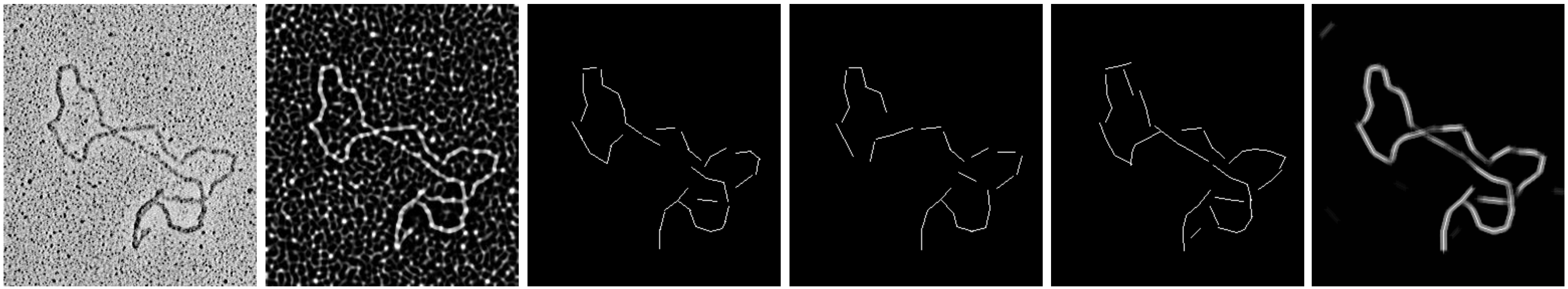
Create line hypotheses



Input Gradient $\mathcal{P}_1$ $\mathcal{P}_2$ $\mathcal{P}_3$ $\mathcal{P}_{\hat{\mathcal{S}}}$

- Stochastic model is sensitive to the selection of hyperparameter
- Learning is NOT feasible
  - Ground truth is given as a binary segmentation map
- To avoid estimating hyperparameter,
  1. We build line hypotheses with respect to ***K* different hyperparameter vectors**
  2. Integrate line hypotheses to reduce sample space
  3. Find the most promising line hypothesis and use its hyperparameter vector

Integrate line hypotheses



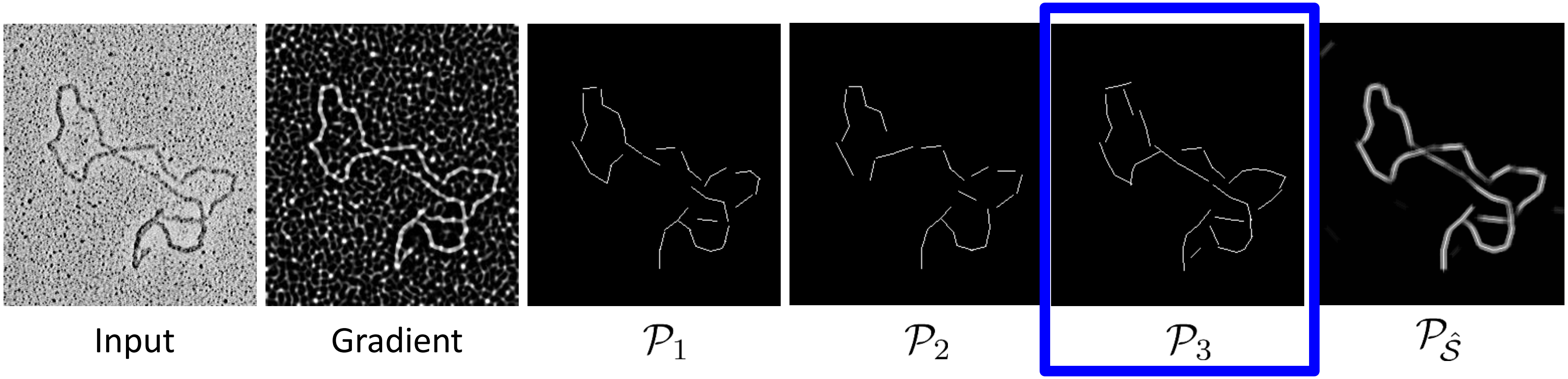
Input Gradient $\mathcal{P}_1$ $\mathcal{P}_2$ $\mathcal{P}_3$ $\mathcal{P}_{\hat{\mathcal{S}}}$

- Assumption
  - Prominent line segment will be observed more frequently
- Mixture density

$$\mathcal{P}_{\hat{\mathcal{S}}} = \frac{1}{K} \sum_{k=1}^{K} \mathcal{P}_k$$

  - Shows consensus between line hypotheses
  - Reduce sampling space
  - Criterion for hyperparameter vector selection

Integrate line hypotheses



- Update data likelihood

$$E'_{\text{data}}(s_i) = E_{\text{data}}(s_i) \boxed{- \log \mathcal{P}_{\hat{\mathcal{S}}}(s_i)}$$

  - Induce reduction of sampling space

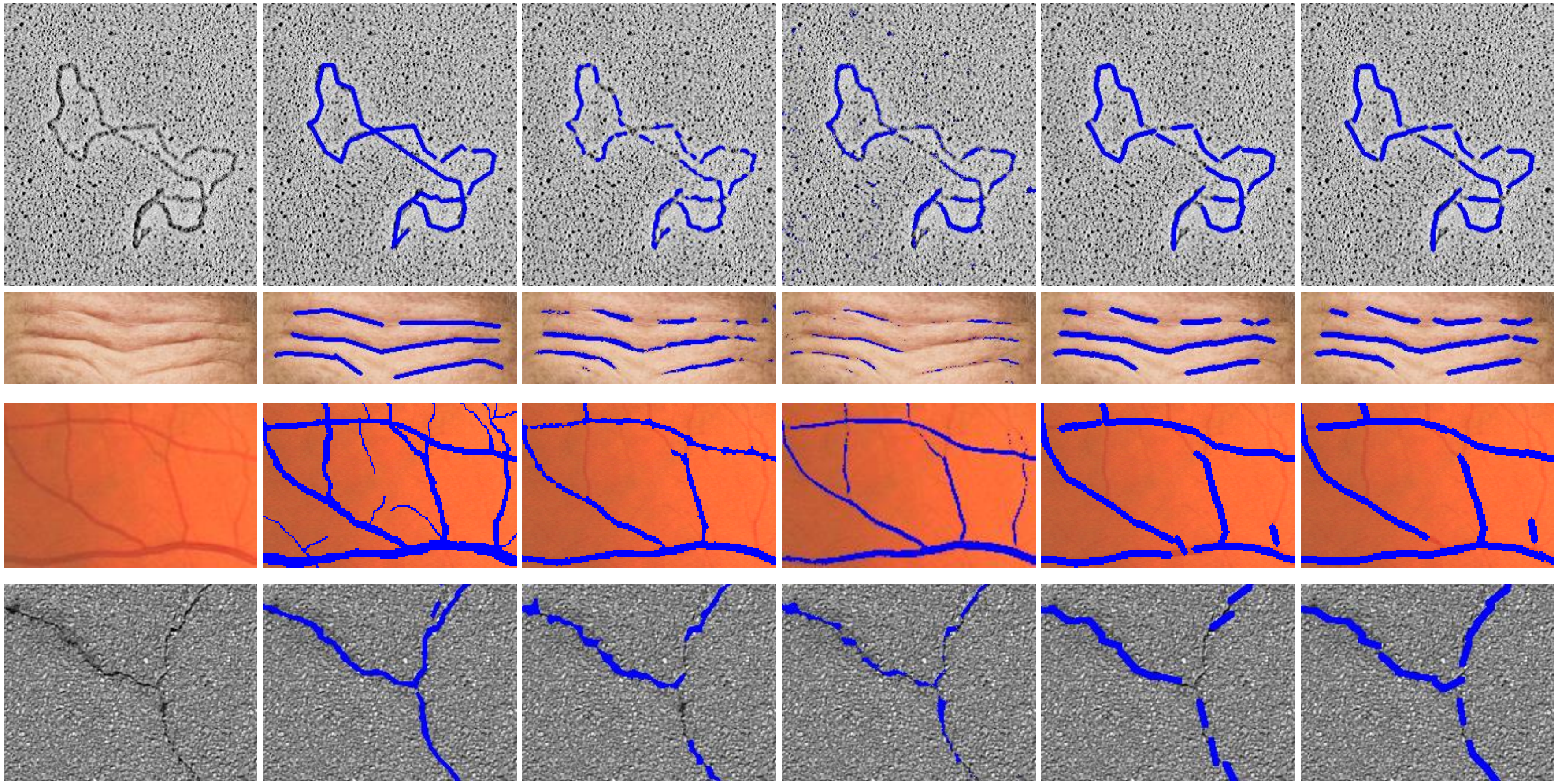
- Find the most promising hyperparameter vector

$$\hat{k} = \underset{k=\{1,\ldots,K\}}{\operatorname{argmax}} \ \mathrm{CC}(\mathcal{P}_{\hat{\mathcal{S}}}, \mathcal{P}_k)$$

- Re-Simulate Markov chain

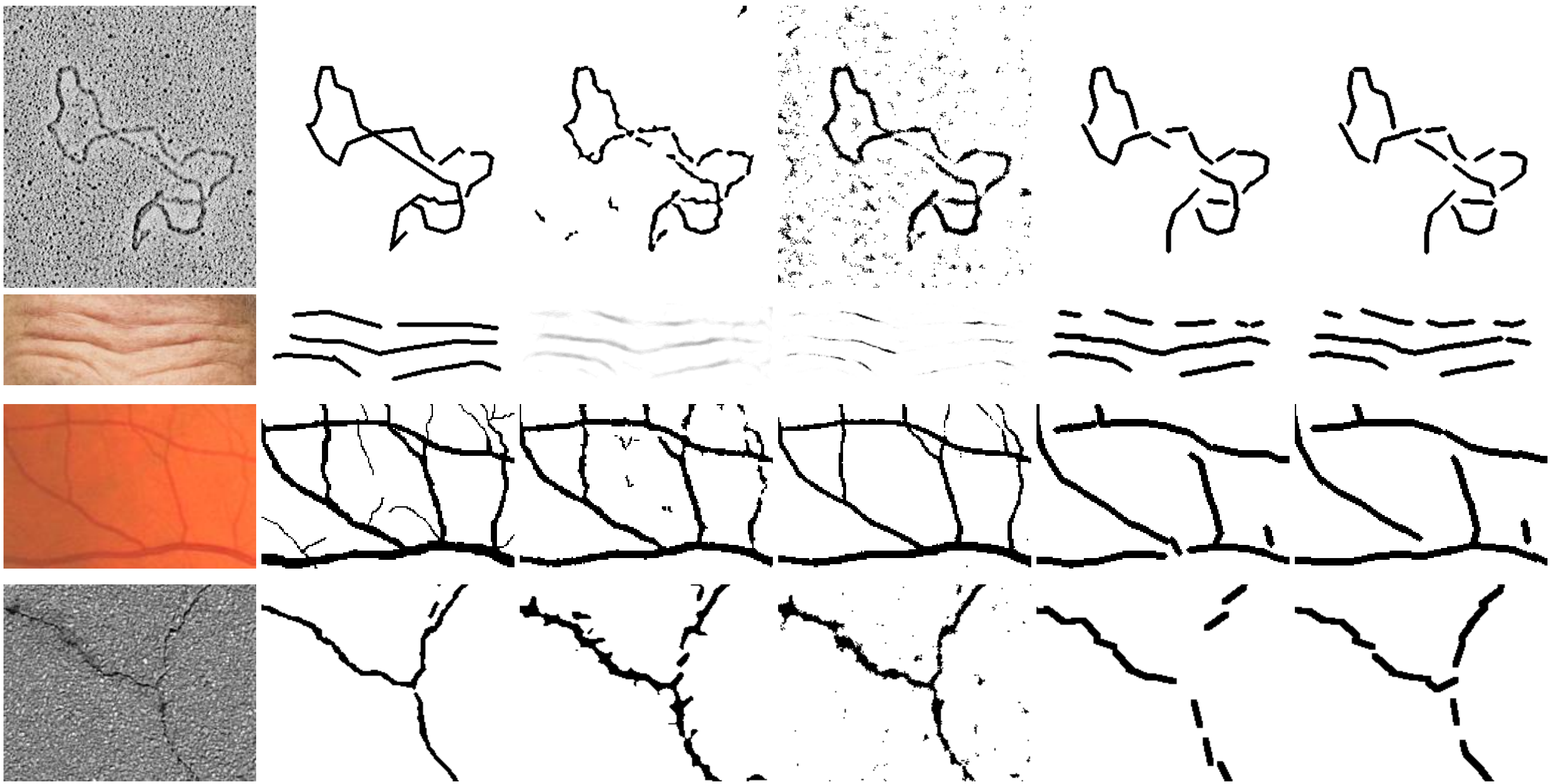
$$\mathbf{s}^* = \underset{\mathbf{s} \subset \mathbb{S}}{\operatorname{argmin}} \sum_{i=1}^{\#(\mathbf{s})} E'_{\text{data}}(s_i) + \sum_{s_m \sim s_n} E_{\text{prior}}(s_m, s_n; \boldsymbol{\omega}_{\hat{k}}).$$

Experimental Results



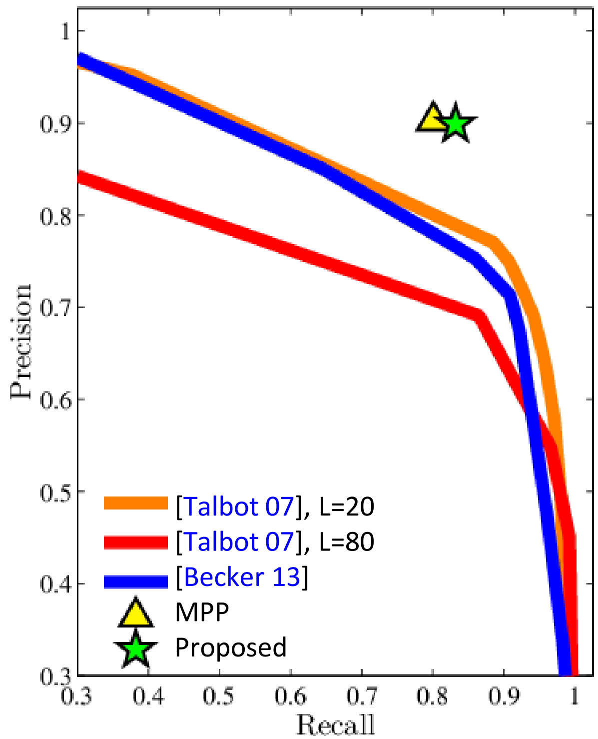
Input | Ground truth | [Talbot 07] | [Becker 13] | MPP | Proposed

Experimental Results

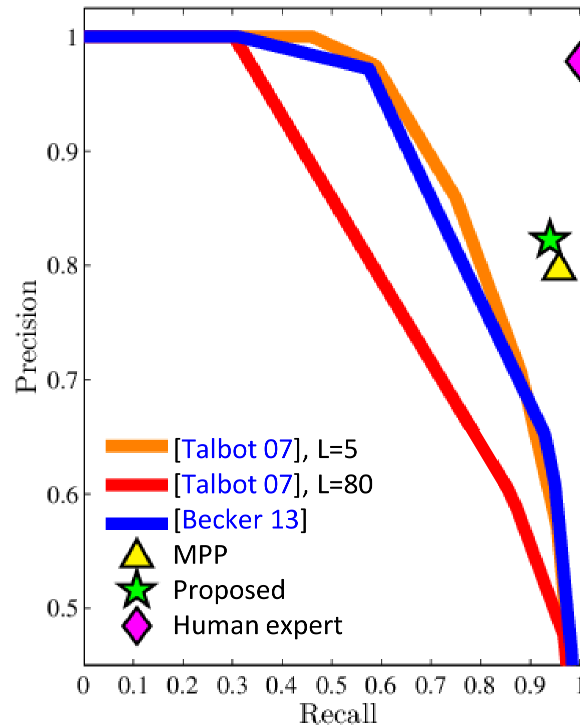


Input Ground truth [Talbot 07] [Becker 13] MPP Proposed

Experimental Results



DNA



Wrinkles

Summary

- Stochastic model for curvilinear structures
  - *e.g., Wrinkles, DNA filaments, road cracks, blood vessels, ...*
  - ***Data term***: *Image gradient & morphological filtering responses*
  - ***Prior term***: *To provide smoothly connected lines*
  - ***Simulation****: RJMCMC with delayed rejection*
- Reduce parameter dependencies of the stochastic modeling with hypotheses integration

- **Limitation**

  Heuristically designed prior energy

  Fails to find varying thicknesses

  Heavy computation

Inference of Curvilinear Structure

Supervised machine learning

- **Goal**: obtain a function $h : \mathcal{Z} \mapsto \mathcal{Y}$ which maps an input space $\mathbf{z} \in \mathcal{Z}$ to an output space $y \in \mathcal{Y}$
- *Supervised machine learning algorithm* evaluates the quality of hypothesis $\hat{y} = h(\mathbf{z}; \mathbf{w})$ with labeled examples

Machine learning vs. Structured learning

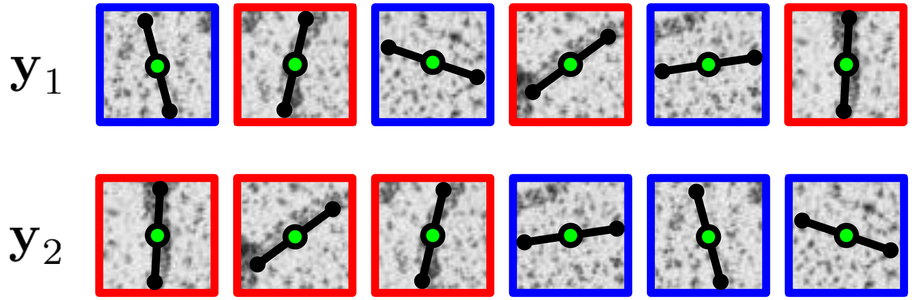
- Inputs can be any kind of objects (**Both**)
- Output is a real number (*Machine learning*)

  e.g., Classification, regression, ...
- Outputs are complex / structured objects (*Structured learning*)

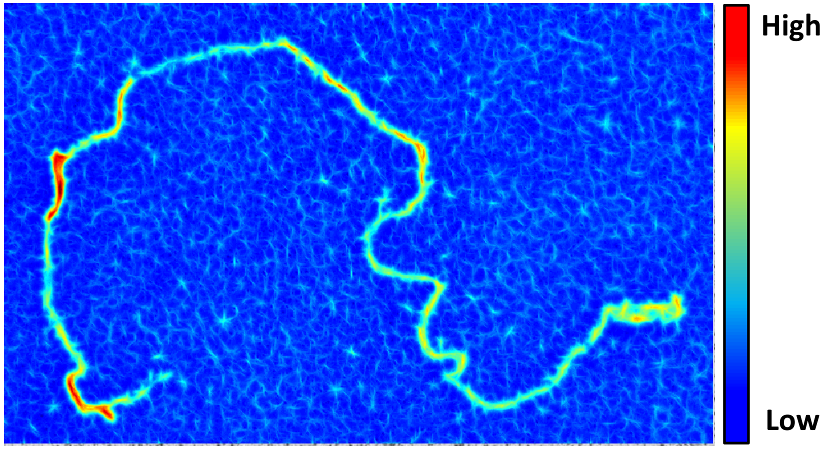
  e.g., Segmentation, protein sequence, NLP, ranking, ...

Structured ranking learning

- Learn a ranking function to evaluate correspondence between line segments and feature maps
- Curvilinear structure will be reconstructed by the high ranked line segments
- Score function $H(\mathbf{s}, \mathbf{y}; \mathbf{w})$



$$H(\mathbf{s}, \mathbf{y}_1; \mathbf{w}) < H(\mathbf{s}, \mathbf{y}_2; \mathbf{w})$$

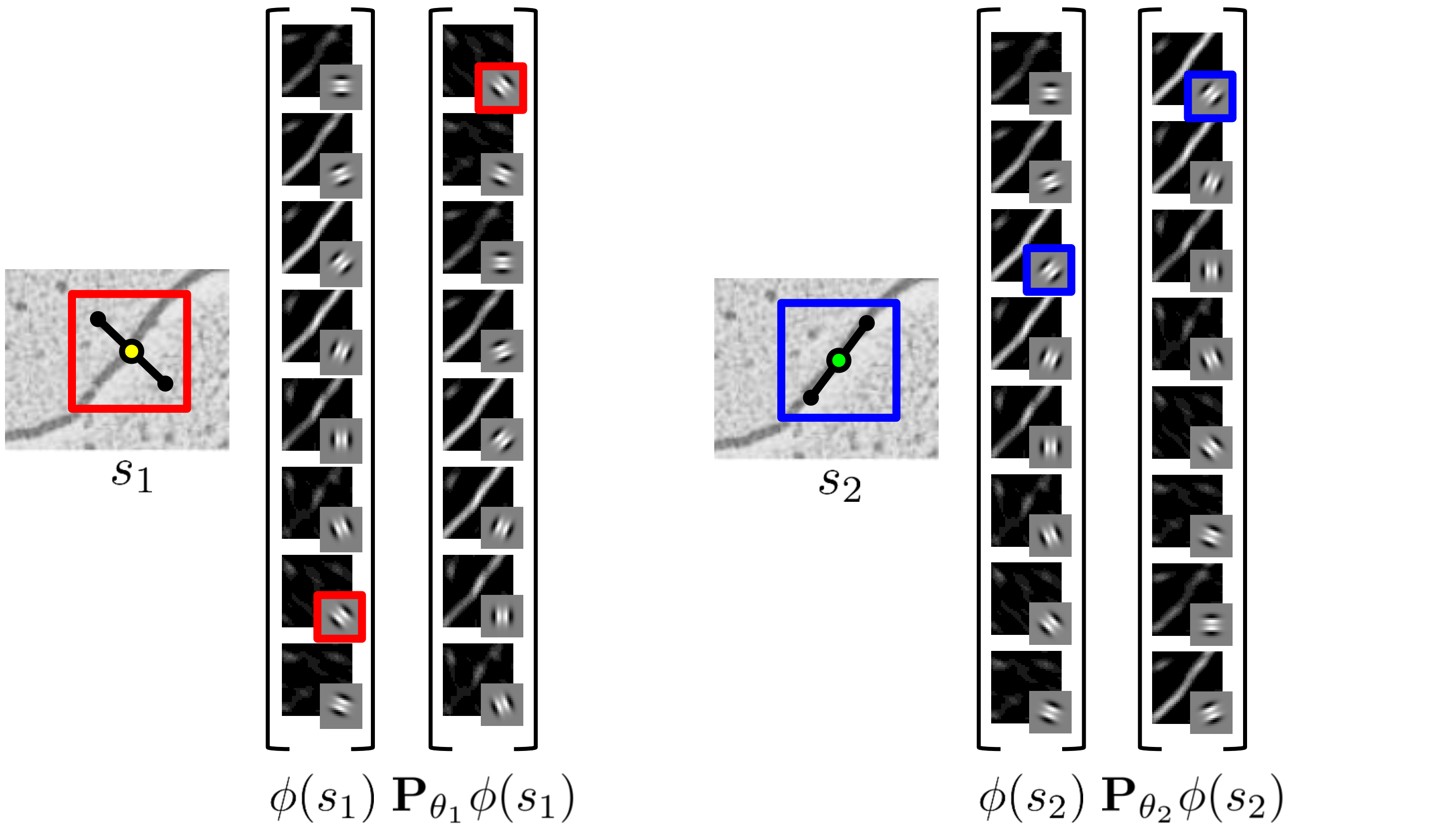


Output ranking scores

- Orientation of line segments?

Orientation-aware curvilinear feature

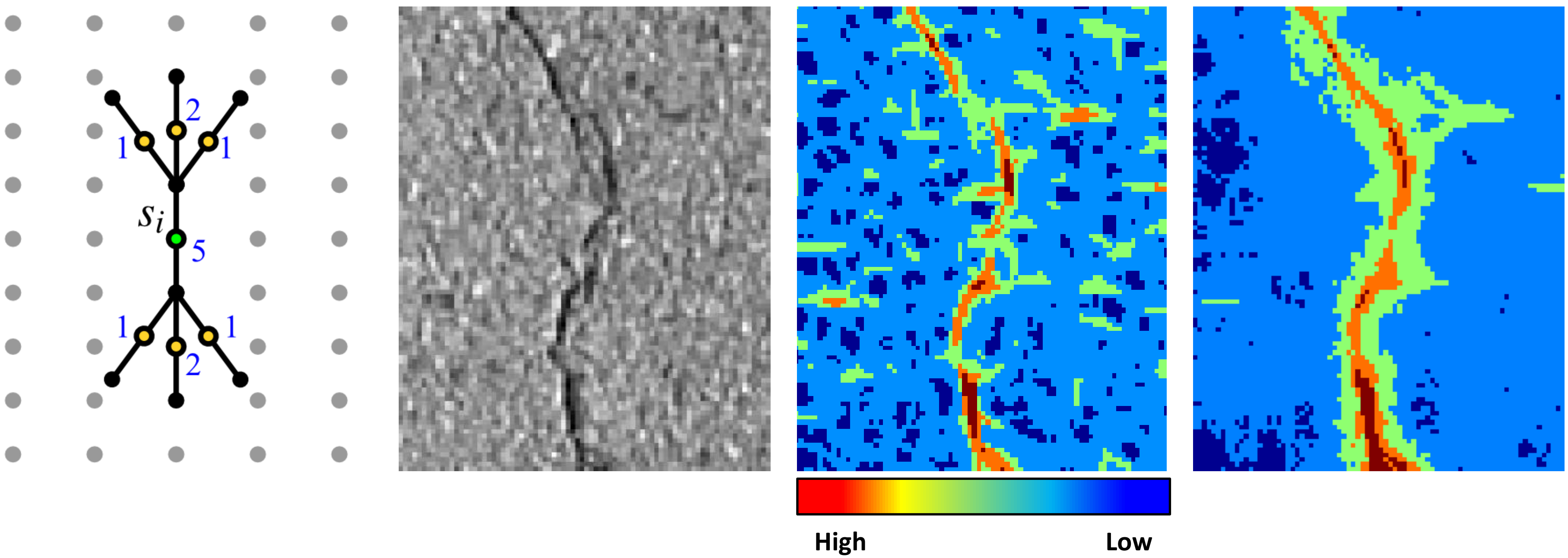
- Model parameter $\mathbf{w}$ determines the relative importance of feature maps
- Permute the elements in the feature vector according to the given orientation



Spatial grouping of the features

- Enhance spatial coherence of the output ranking scores
- Weighted sum with neighboring set of line segments

$$\bar{\phi}(s_i) = \frac{\sum_{j \in \mathcal{N}_i} \omega_j \phi'(s_j)}{\sum_{j \in \mathcal{N}_i} \omega_j}$$



Learning

- We need a training dataset $\mathcal{D} = \{(s_i, y_i)\}_{i=1}^{K}$
  - A list of line segments (**Easy**)
  - The relevant raking values (**?**)
- Ground truth (GT) is given as a binary segmentation map
  - No shape information w.r.t. line segments, *i.e., length, orientation, thickness*
- Evaluate the shape dissimilarity (F_1) between the line segment and the corresponding image patch from ground truth

Image	GT	0^o	22.5^o	45^o	67.5^o	90^o	112.5^o	135^o	157.5^o
	F_1	0.56	**0.78**	0.51	0.30	0.27	0.22	0.19	0.26
	Ranking	2	**1**	3	4	5	7	8	6

True positive　False positive　False negative

Learning

- Prediction is performed by finding rankings that maximize the score function

$$\hat{\mathbf{y}} = \operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}} H(\mathbf{s},\mathbf{y};\mathbf{w}) = \operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}} \mathbf{w}^\intercal \Psi(\mathbf{s},\mathbf{y})$$

- Joint feature map

$$\Psi(\mathbf{s},\mathbf{y}) = \sum_{ij} y_{ij} \left(\bar{\phi}(s_i) - \bar{\phi}(s_j)\right),$$

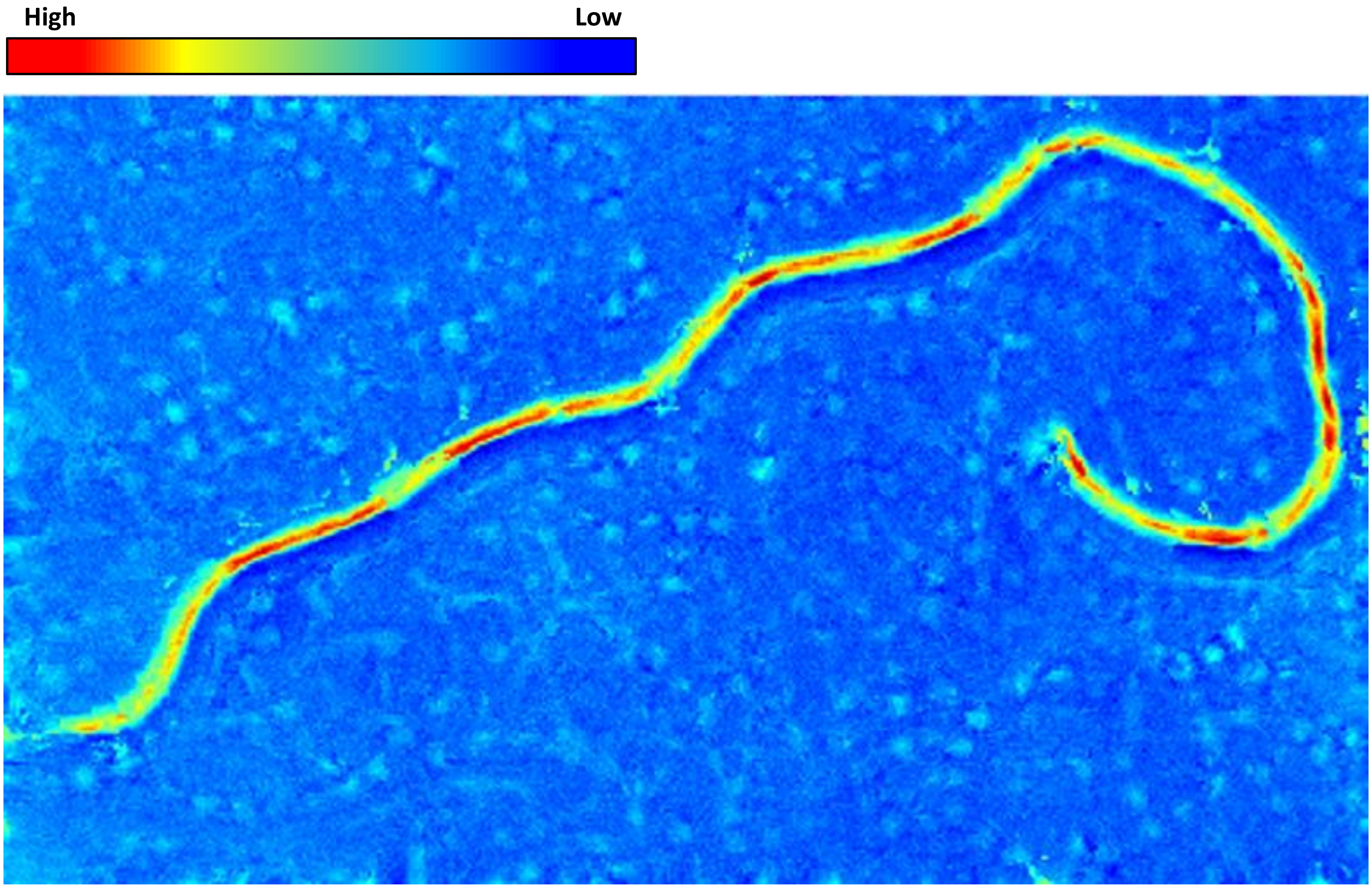
- Ranking matrix

$$y_{ij} = \begin{cases} +1 & \text{if } y_i > y_j, \\ -1 & \text{otherwise.} \end{cases}$$

- Optimize constrained objective function via cutting plane algorithm [Joachims 09]

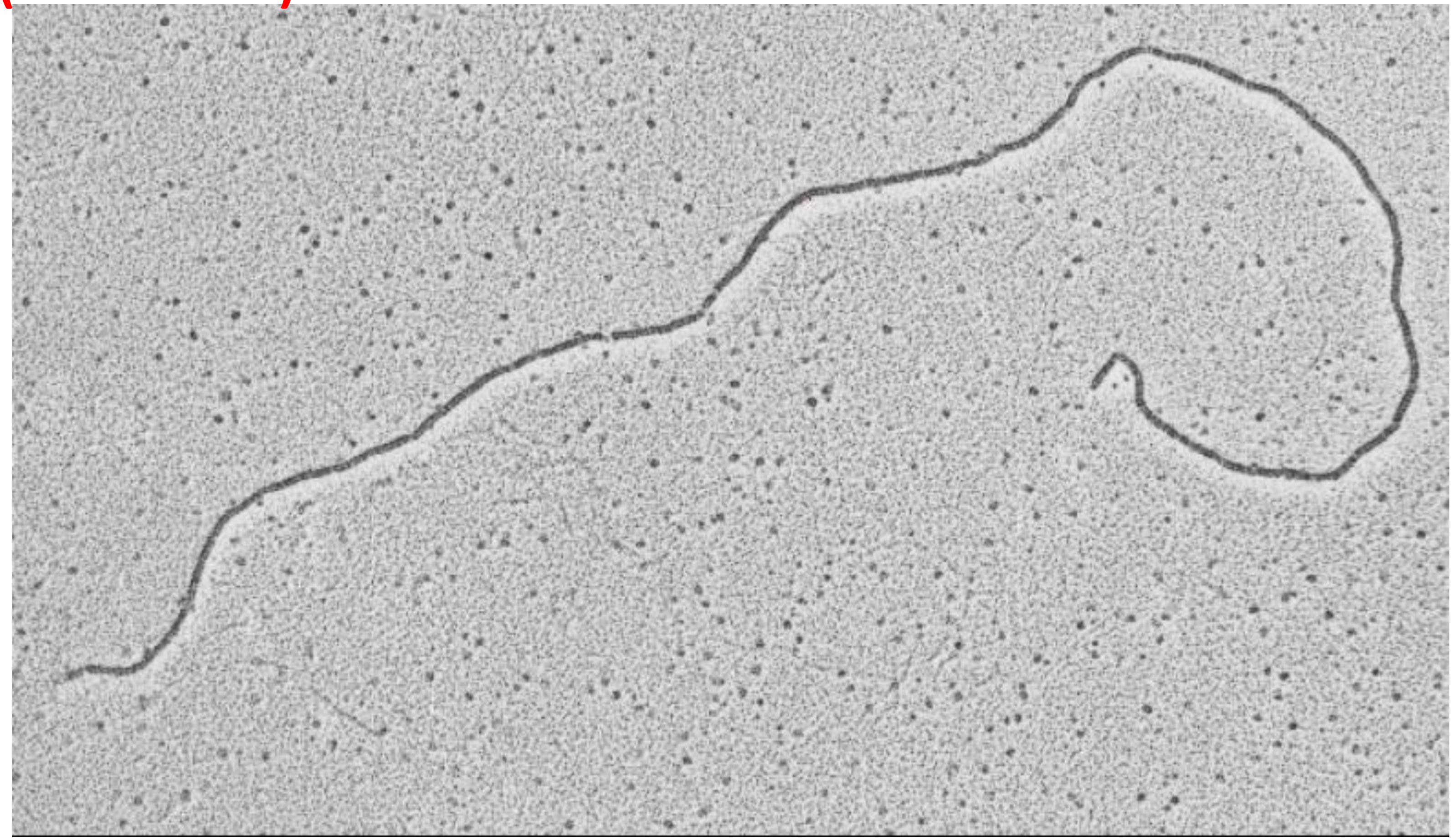
Output ranking score map

- Output ranking scores highlight the latent curvilinear structure



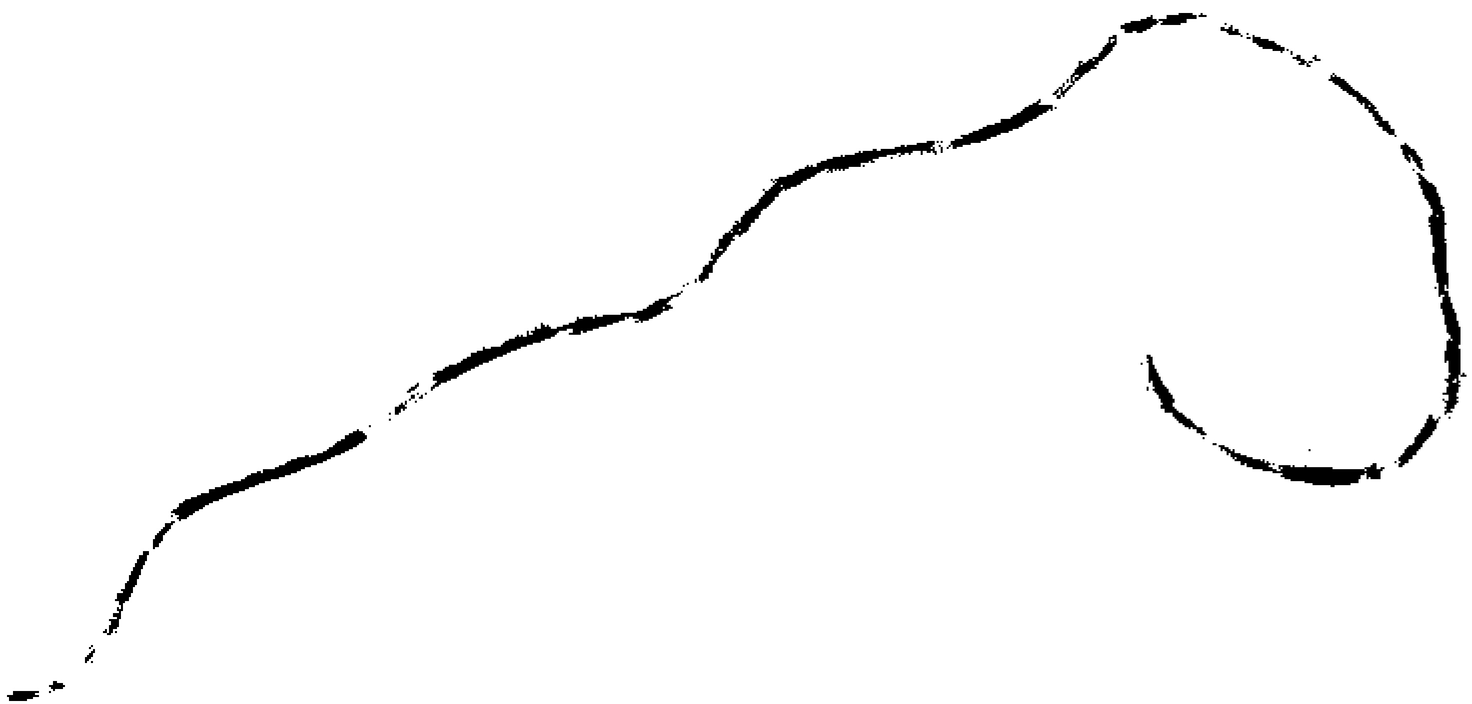
Binary segmentation map

- Compute the average proportion of pixels being part of the curvilinear structure from training images for **stop criterion (≈threshold)**



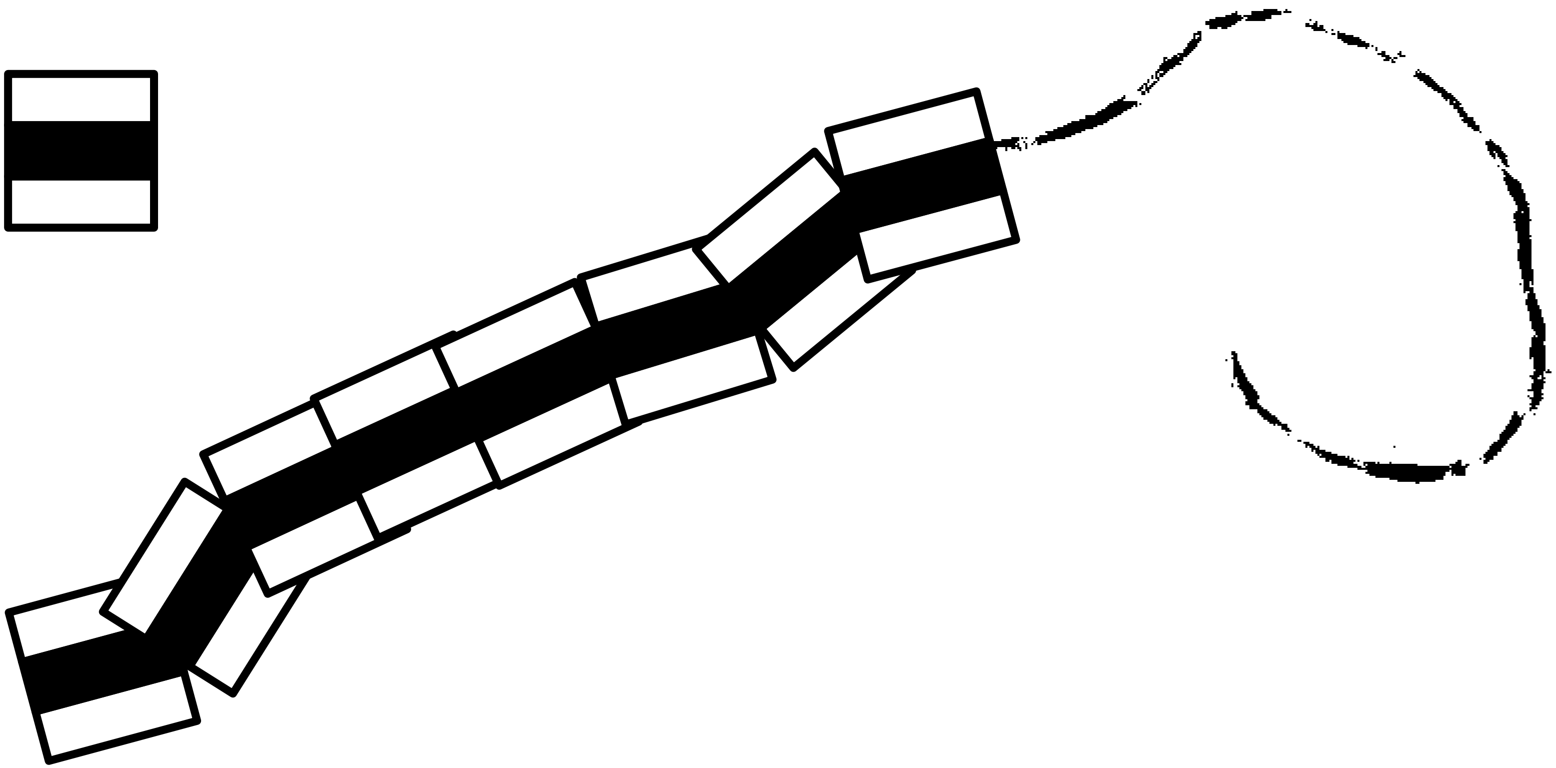
Binary segmentation map

- Remain pixels according to the output rankings
- Topology can be broken



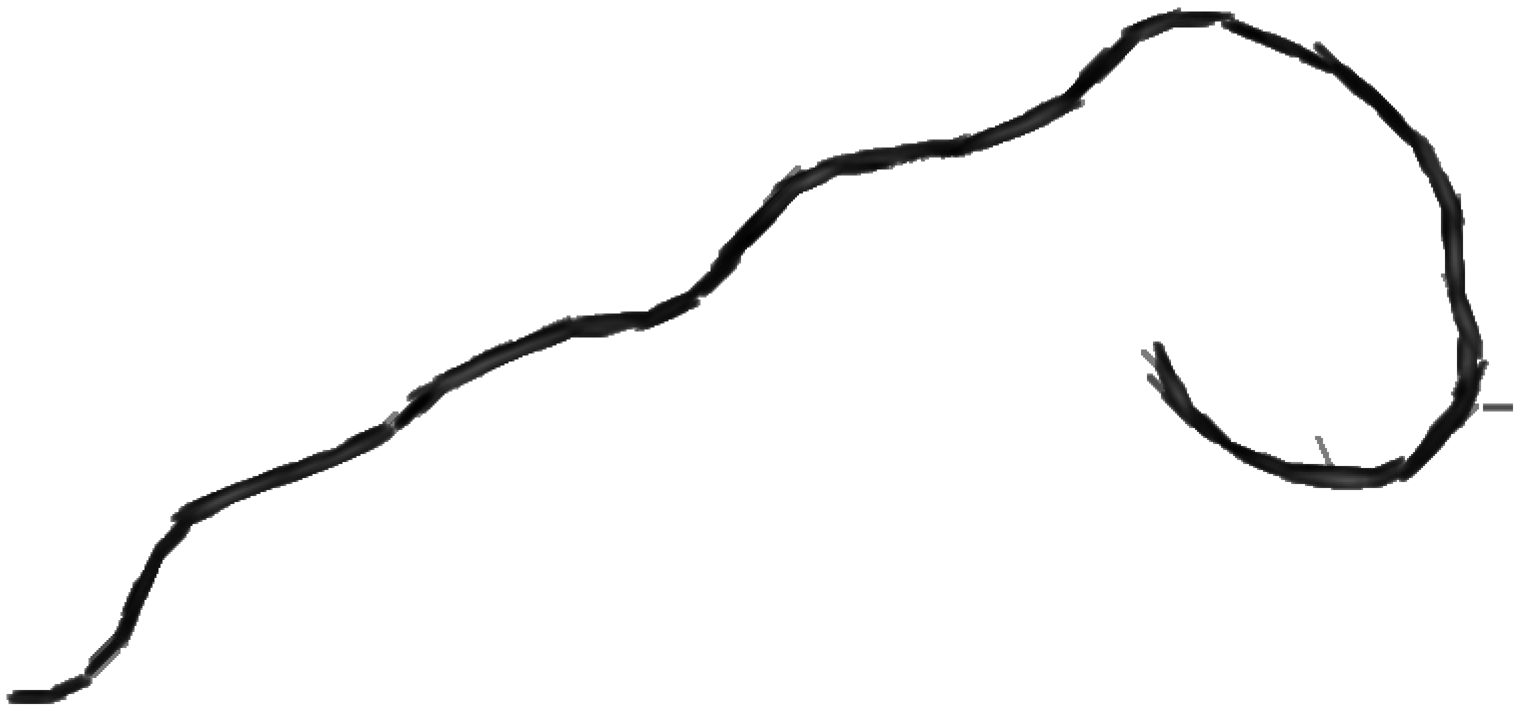
Dissimilarity score map

- Each pixel encodes shape information (length, orientation, and thickness)



Dissimilarity score map

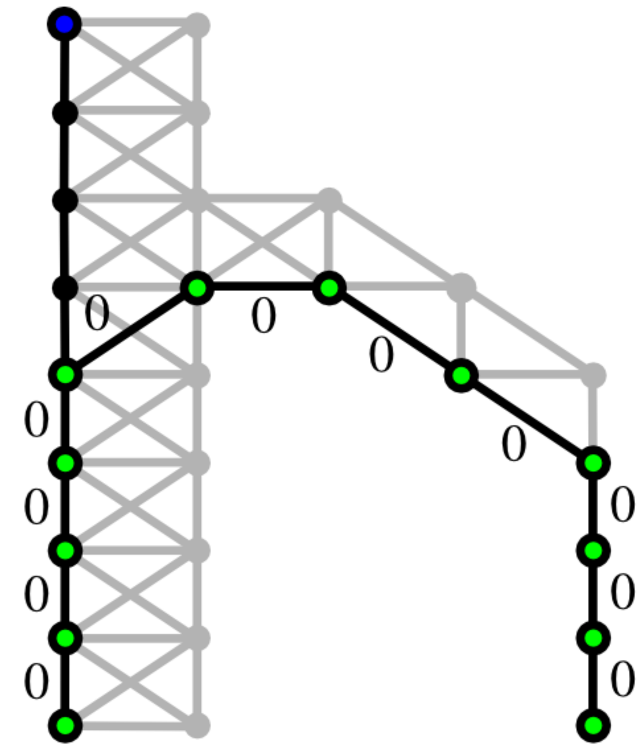
- Values are used to generate a graph for curvilinear path reconstruction



Progressive curvilinear path reconstruction

1. Induce a subgraph **G′** using structured output ranking scores
2. Randomly select a vertex **t** and find the longest geodesic path (**t** → **u**)
3. Find the longest geodesic path from **u** to a vertex **v** at the maximum distance from **u**

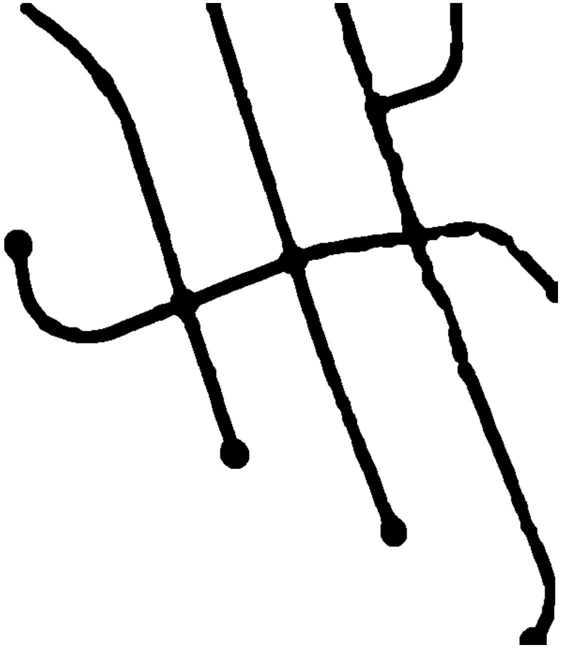
   The distance of path **u** → **v** is a diameter of **G′**
4. Assign 0 weight for all edges on this path (**u** → **v**)
5. Repeat the process step 2 to 4 to add branches which are longer than pre-defined length



Progressive curvilinear path reconstruction

- Iteratively find the longest geodesic path in the graph
- Can illustrate topological features in different levels of detail

Input image



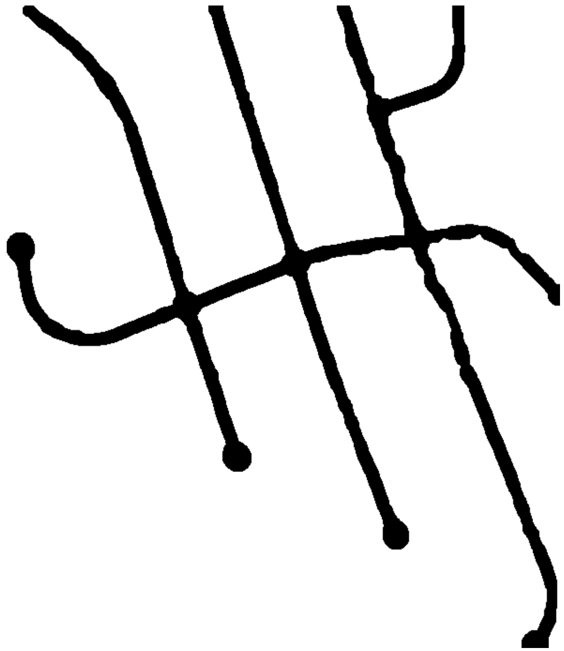
Ground truth

Proposed

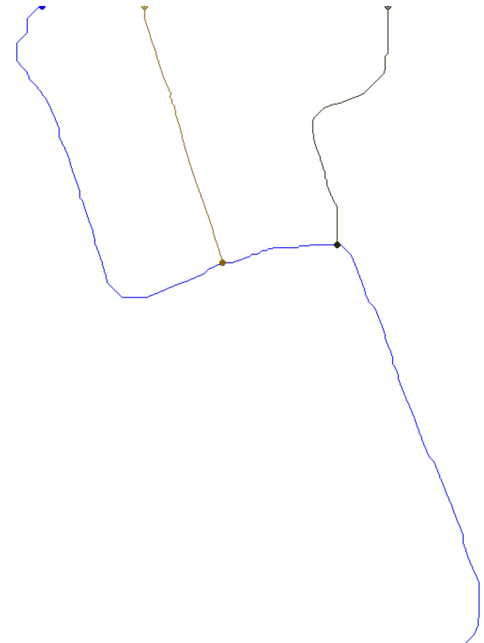
Progressive curvilinear path reconstruction

- Iteratively find the longest geodesic path in the graph
- Can illustrate topological features in different levels of detail

Input image



Ground truth

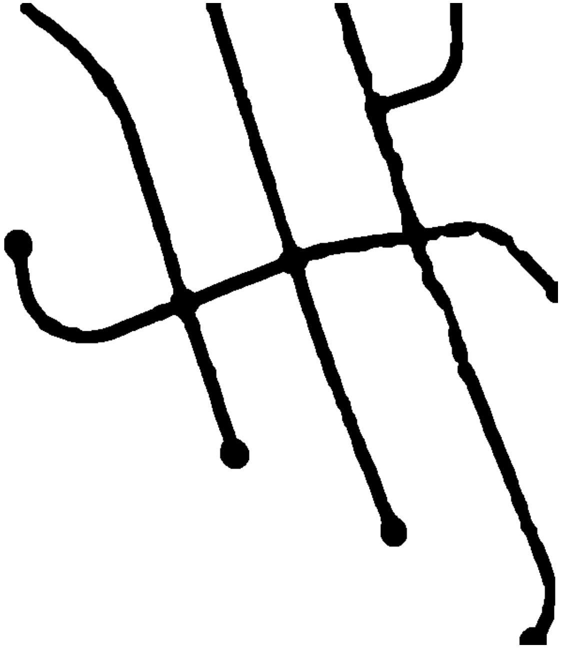


Proposed

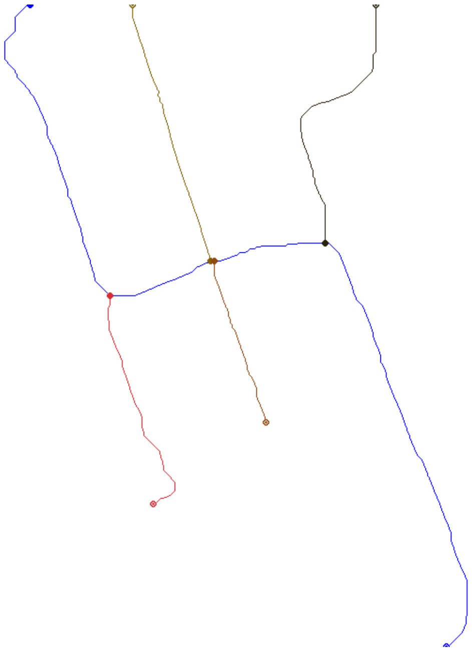
Progressive curvilinear path reconstruction

- Iteratively find the longest geodesic path in the graph
- Can illustrate topological features in different levels of detail

Input image



Ground truth

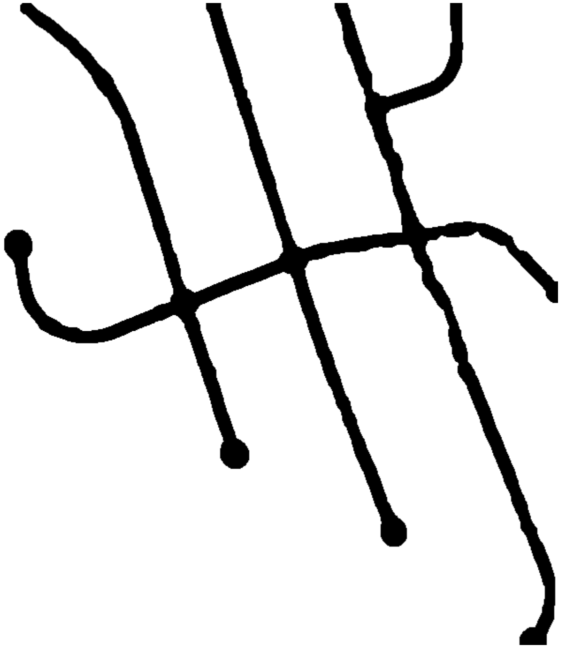


Proposed

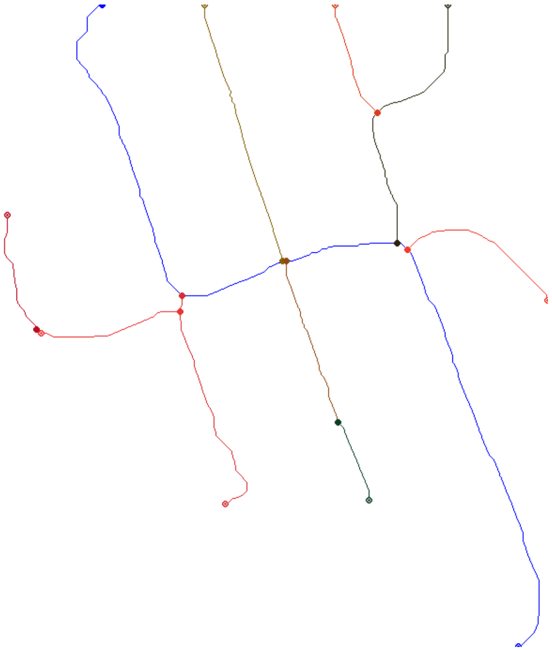
Progressive curvilinear path reconstruction

- Iteratively find the longest geodesic path in the graph
- Can illustrate topological features in different levels of detail

Input image



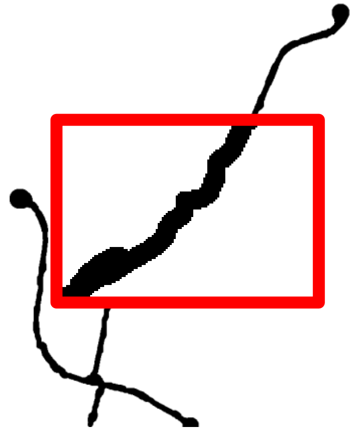
Ground truth



Proposed

Experimental results: Aerial

Input

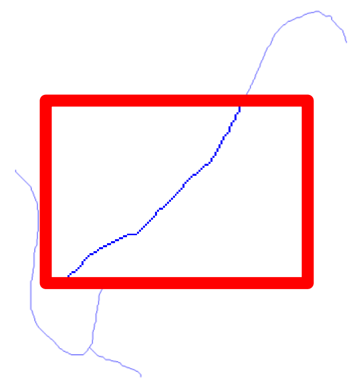


Ground truth

[Law 08]

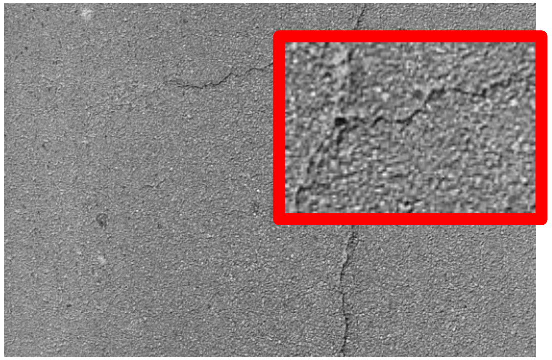
[Becker 13]

[Sironi 14]

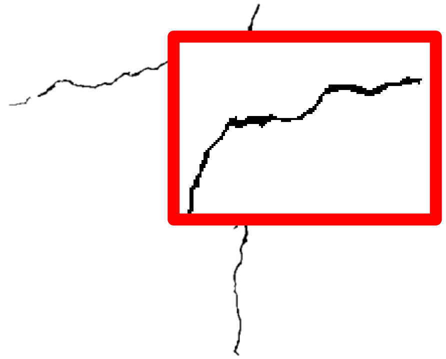


Proposed

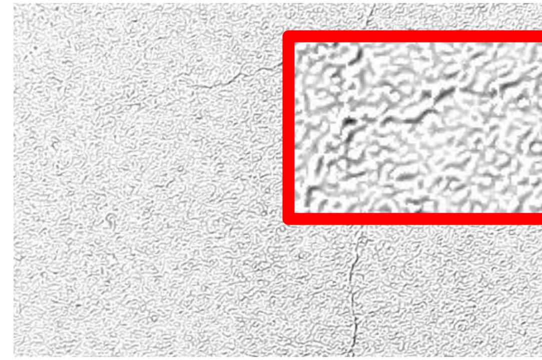
Experimental results: Cracks



Input

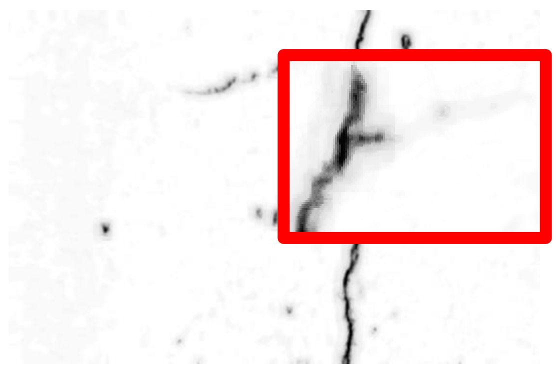


Ground truth

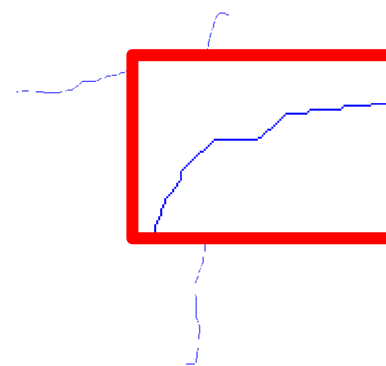


[Law 08]

[Becker 13]



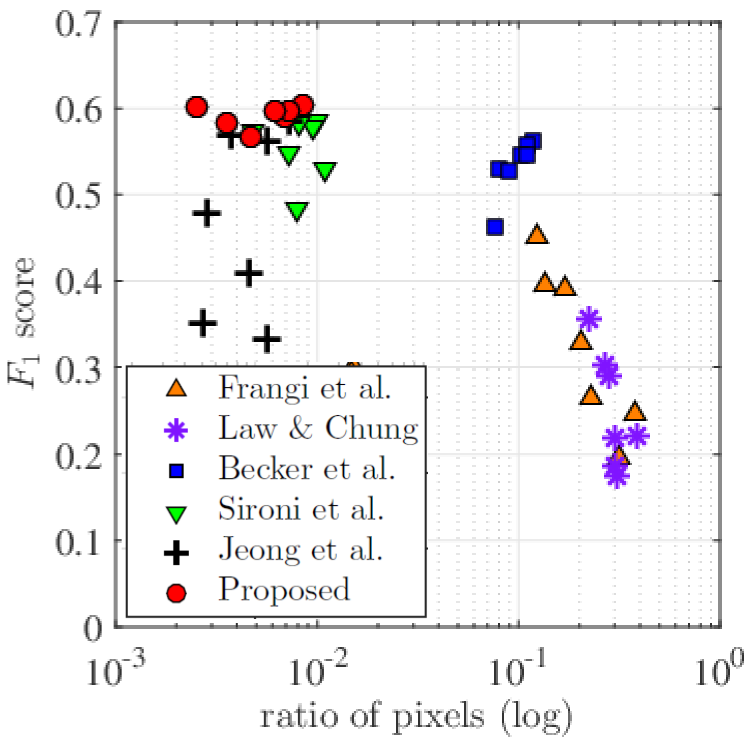
[Sironi 14]



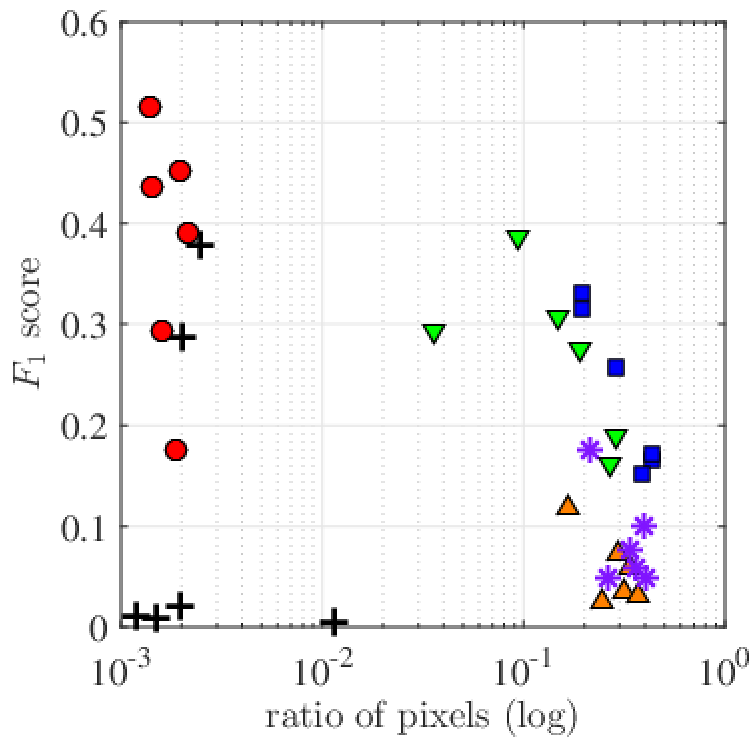
Proposed

Experimental results

- Proposed curvilinear path reconstruction shows the highest F_1 performance with the minimum number of pixels



Aerial



Cracks

Summary

- Structured ranking learning for curvilinear structures
  - ***Learn a ranking function****: to evaluate correspondence between line segments and feature maps*
  - ***Orientation-aware feature vector****: permute elements in feature vector according to the given orientation*
  - ***Structured SVM****: employed to obtain model parameter*
- Progressive curvilinear path reconstruction shows topological features in different levels of detail

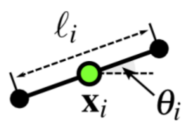
- **Limitation**

  Ambiguous stopping criterion

  Fails to interpret varying thicknesses

  Few parameters are manually chosen, *e.g., pre-defined length*

Conclusions

- Generic curvilinear structure reconstruction models
  - *Retina, DNA filament, Road, Cracks, ...*
- Curvilinear features
  - Image gradients
  - Morphological filtering
- Line segment $s_i = (\mathbf{x}_i, \ell_i, \theta_i)$ 

- Find an optimal set of line segments
- Stochastic modeling
  - Maximize a posterior probability of lines segments for given image
- Structured ranking learning
  - Learn a function to evaluate the correspondence between lines and feature maps
- Progressive curvilinear path reconstruction algorithm
  - Provides topological features in different levels of detail

Perspectives

- Multiscale approach
  - To take into account varying thickness
- Stochastic model
  - Hierarchical modeling
  - Bezier curve [Bama 15]
- Structured ranking learning
  - Employing non-line shape templates (bifurcation patterns [Azzopardi 11]) to evaluate rankings
- Speed up
  - Applying parallel MCMC sampler [Verdie 14]

Thank you!

References

[Frangi 98] A. F. Frangi, W. J. Niessen, K. L. Vincken & M. A. Viergever. Multiscale vessel enhancement filtering. In MICCAI, pages 130–137, 1998.

[Staal 04] J. J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever & B. van Ginneken. Ridge based vessel segmentation in color images of the retina. IEEE TMI, vol. 23, no. 4, pages 501–509, 2004.

[Law 08] M. W. Law & A. Chung. Three dimensional curvilinear structure detection using optimally oriented flux. In ECCV, pages 368–382, 2008.

[Zhao 11] T. Zhao, J. Xie, F. Amat, N. Clack, P. Ahammad, H. Peng, F. Long & E. Myers. Automated reconstruction of neuronal morphology based on local geometrical and global structural models. Neuroinformatics, vol. 9, no. 2–3, pages 247–261, 2011.

[Wang 11] Y. Wang, A. Narayanaswamy & B. Roysam. Novel 4D open-curve active contour and curve completion approach for automated tree structure extraction. In CVPR, pages 1105–1112, 2011.

[Peng 11] H. Peng, F. Long & G. Myers. Automatic 3D neuron tracing using all-path pruning. Bioinformatics, vol. 27, no. 13, pages 239–247, 2011.

[Turetken 13] E. Türetken, C. Becker, P. Glowacki, F. Benmansour & P. Fua. Detecting irregular curvilinear structures in gray scale and color imagery using multi-directional oriented flux. In ICCV, pages 1553–1560, 2013.

References

[Batool 12] N. Batool & R. Chellappa. *Modeling and detection of wrinkles in aging human faces using marked point processes*. In ECCV Ws/Demos, pages 178–188, 2012.

[Jeong 14] S.-G. Jeong, Y. Tarabalka & J. Zerubia. Marked point process model for facial wrinkle detection. In ICIP, pages 1391–1394, 2014.

[Lacoste 05] C. Lacoste, X. Descombes & J. Zerubia. Point processes for unsupervised line network extraction in remote sensing. IEEE TPAMI, vol. 27, no. 10, pages 1568–1579, 2005.

[Hu 07] J. Hu, A. Razdan, J. C. Femiani, M. Cui & P. Wonka. Road network extraction and intersection detection from aerial images by tracking road footprints. IEEE TGRS, vol. 45, no. 12, pages 4144–4157, 2007.

[Valero 10] S. Valero, J. Chanussot, J. A. Bendiktsson, H. Talbot & B. Waske. Advanced directional mathematical morphology for the detection of the road network in very high resolution remote sensing images. Pattern Recognition Lett., vol. 31, no. 10, pages 1120–1127, 2010.

[Iyer 05] S. Iyer & S. Sinha. A robust approach for automatic detection and segmentation of cracks in underground pipeline images. Image and Vision Computing, vol. 23, no. 10, pages 921–933, 2005.

[Azzopardi 11] G. Azzopardi & N. Petkov, Detection of retinal vascular bifurcations by trainable V4-like filetrs, In CAIP, pages 451-459, 2011

References

[Chambon 10] S. Chambon, C. Gourraud, J.-M. Moliard & P. Nicolle. Road crack extraction with adapted filtering and Markov model-based segmentation. In VISAPP(2), pages 81–90, 2010.

[Green 95] P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika, vol. 82, no. 4, pages 771–732, 1995.

[Green 01] P. J. Green & A. Mira. Delayed rejection in reversible jump Metropolis-Hastings. Biometrika, vol. 88, no. 4, pages 1035–1053, 2001.

[Kirkpatrick 83] S. Kirkpatrick, C. D. Gelatt & M. P. Vecchi. Optimization by Simulated Annealing. Science, vol. 220, no. 4598, pages 671–680, 1983.

[Joachims 09] T. Joachims, T. Finley & C.-N. Yu. Cutting-plane training of structural SVMs. Machine Learning, vol. 77, no. 1, pages 27–59, 2009.

[Corneil 01] D. G. Corneil, F. F. Dragan, M. Habib & C. Paul. Diameter determination on restricted graph families. Discrete Applied Mathematics, vol. 113, no. 2–3, pages 143–166, 2001.

[Verdie 14] Y. Verdié & F. Lafarge. Detecting parametric objects in large scenes by Monte Carlo sampling. IJCV, vol. 106, no. 1, pages 57–75, 2014.

[Bama 15] K. Bama, T. Sziranyi, M. Borda, & O. Lavialle, Marked point processes for enhancing seismic fault patterns, Journal of Applied Geophysics no. 118, pages 115–123, 2015