大学共同利用機関法人 情報・システム研究機構
国立情報学研究所
National Institute of Informatics

Continuous Domain Analysis of Graph Laplacian Regularization for Image Denoisng

Presenter: Gene Cheung

National Institute of Informatics

[1] Jiahao Pang, Gene Cheung, Wei Hu, Oscar C. Au, "**Redefining Self-Similarity in Natural Images for Denoising Using Graph Signal Gradient**," in *APSIPA ASC*, Siem Reap, Cambodia, December, 2014.

Outline

- Introduction
- Convergence of the Graph Laplacian Regularizer
- Justification of the Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal Graph Laplacian Regularizer
- Conclusion

Lena, $\sigma = 30$

Outline

- Introduction
- Convergence of the Graph Laplacian Regularizer
- Justification of the Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal Graph Laplacian Regularizer
- Conclusion

Lena, $\sigma = 30$

Motivation (I)

- Image denoising—a basic restoration problem:

observation → $$\mathbf{y} = \mathbf{x} + \mathbf{e}$$ ← noise, ← desired signal

- It is under-determined, needs image priors for regularization

fidelity term → $$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \ \mathrm{prior}(\mathbf{x})$$ ← prior term

- Graph Laplacian regularizer: should be small for target patch $\mathbf{x}$

$$S_{\mathrm{G}}(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x} \qquad \mathbf{L} = \mathbf{D} - \mathbf{A}$$ ← graph Laplacian matrix

- Many works use Gaussian kernel to compute graph weights [2]:

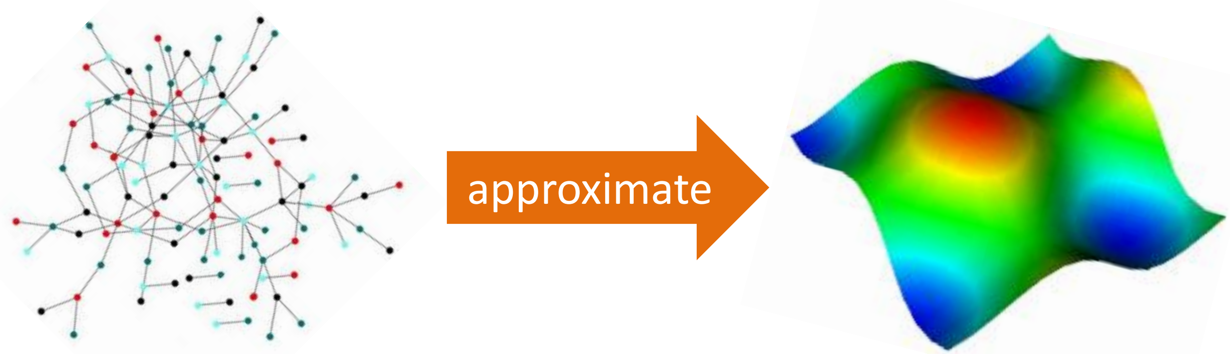
$$w_{ij} = \exp\left(\frac{dist(i,j)^2}{\sigma^2}\right)$$

$\mathrm{dist}(i, j)$ is some distance metric between pixels i and j

[2] D. Shuman, S. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "**The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains**," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

Motivation (II)

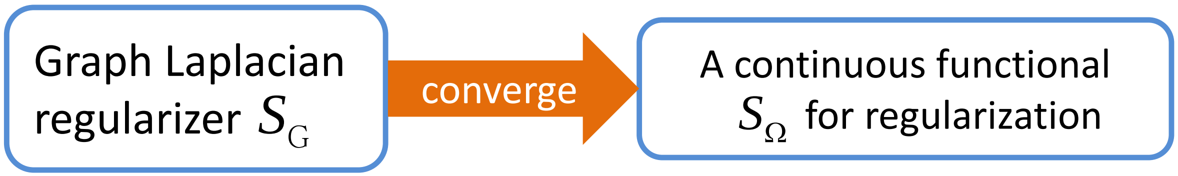
- However…
  a. Why is $S_G(\mathbf{x}) = \mathbf{x}^T\mathbf{L}\mathbf{x}$ a good prior?
  b. Why using Gaussian kernel for edge weights?
  c. How to design a discriminant $\mathbf{x}^T\mathbf{L}\mathbf{x}$ for restoration?
- We answer these by viewing
  - discrete graph as **samples** of high-dimensional manifold.



discrete graph continuous manifold

Our Contributions

1. Using Gaussian kernel to compute graph weights, $S_G(\mathbf{x}) = \mathbf{x}^T \mathbf{L} \mathbf{x}$ converges to a continuous functional S_Ω, which can be interpreted as regularizer in continuous domain.



2. Analysis of functional S_Ω provides understanding of what signals are being discriminated and to what extent, on a point-by-point basis in the continuous domain.

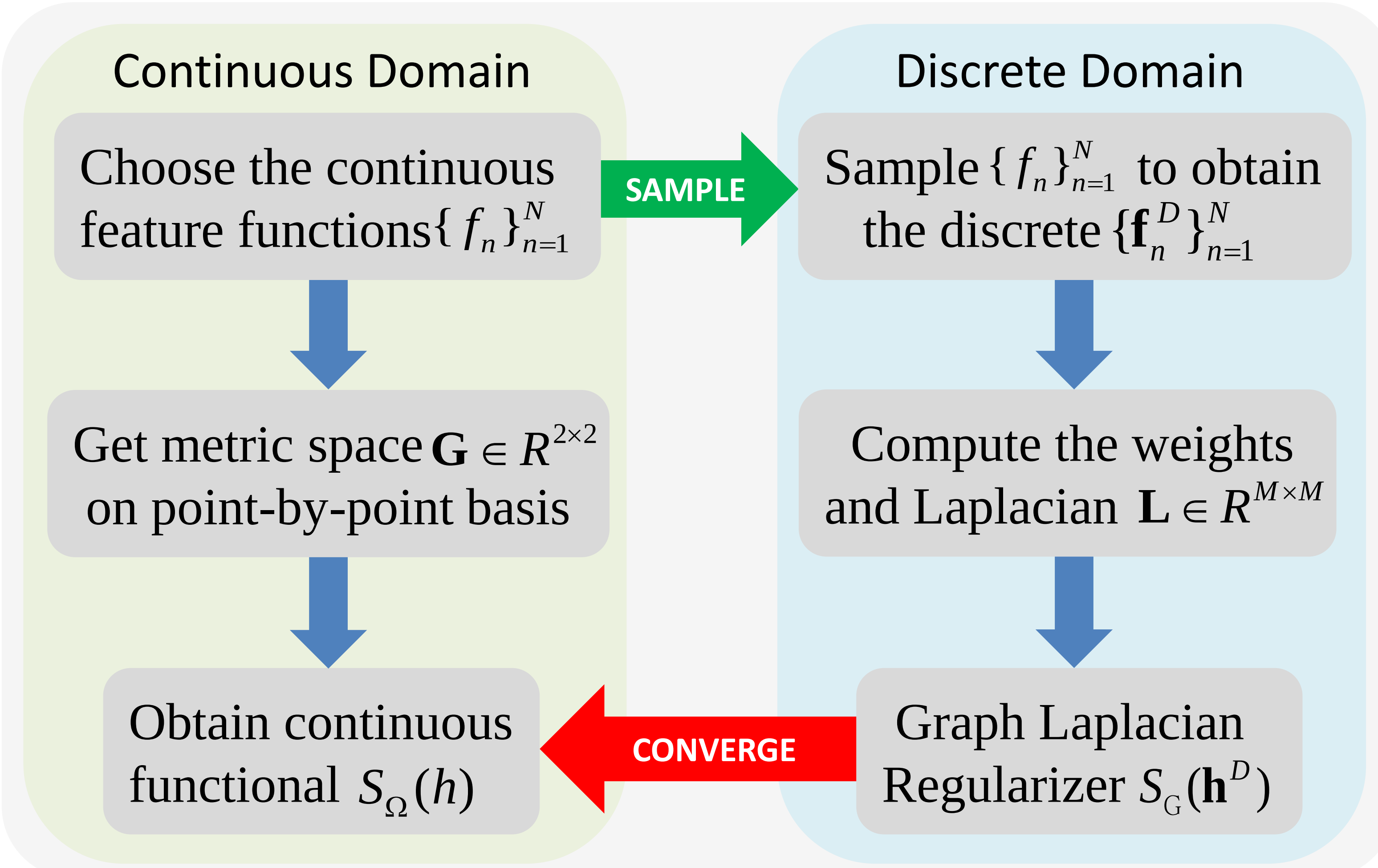
3. We design a discriminant S_Ω for regularization in continuous domain, then obtain the graph Laplacian regularizer S_G

Outline

- Introduction
- Convergence of the Graph Laplacian Regularizer
- Justification of the Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal Graph Laplacian Regularizer
- Conclusion

Lena, $\sigma = 30$

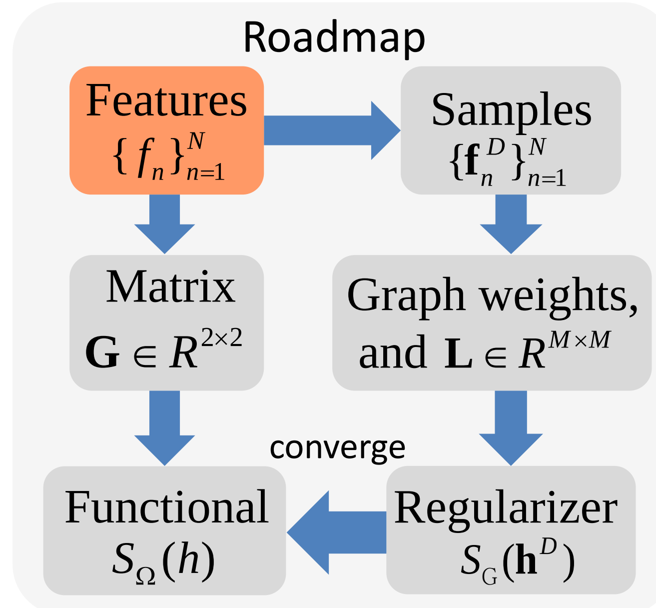
Road Map



- Different $\{f_n\}_{n=1}^N$ leads to different regularization behavior!

Graph Construction (I)

- First, define:
  - 2D domain $\Omega \subset R^2$
    — the shape of an image
  - $\Gamma = \left\{ \mathbf{s}_i = [x_i \ y_i]^{\mathrm{T}} \mid \mathbf{s}_i \in \Omega, 1 \le i \le M \right\}$
    — a set of M random samples uniformly distributed on Ω, construed as pixel locations

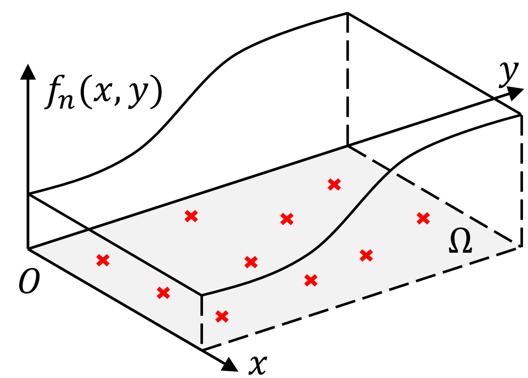


- (Freely) choose N continuous functions

$$f_n(x, y) : \Omega \to R, \ 1 \le n \le N$$

called feature functions, can be

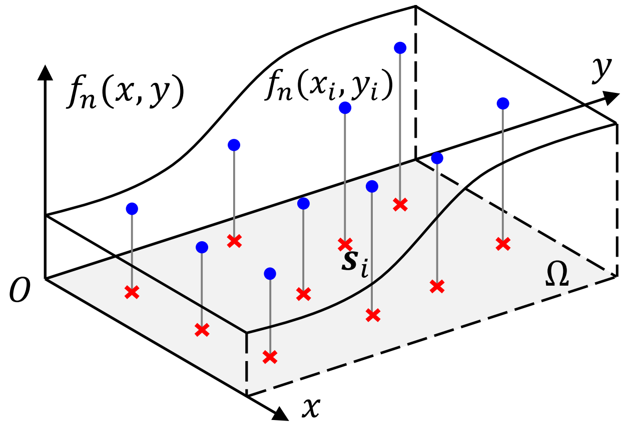
  - intensity for gray-scale image $(N = 1)$
  - R, G, B channels for color image $(N = 3)$

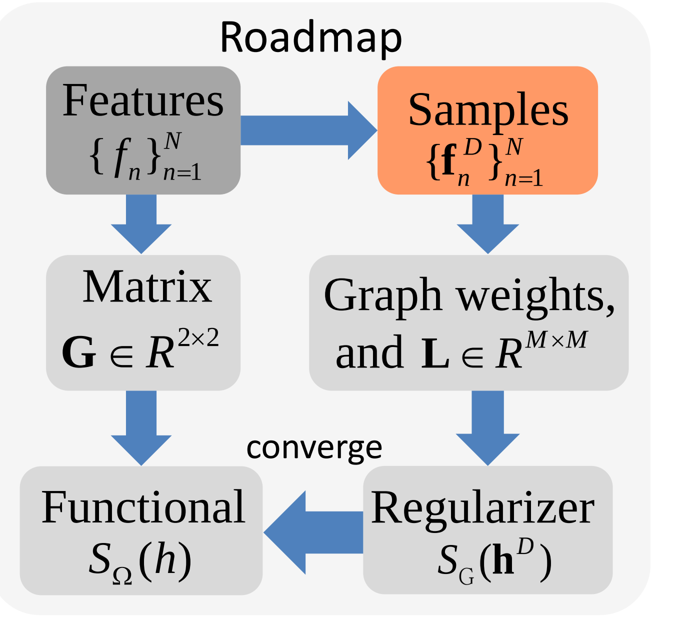


Graph Construction (II)

- Sampling f_n at positions in Γ gives N discretized feature functions

$$\mathbf{f}_n^D = [\, f_n(x_1, y_1)\ f_n(x_2, y_2) \ldots f_n(x_M, y_M)]^{\mathrm{T}}$$





- For each pixel location $\mathbf{s}_i \in \Gamma$, define a length $N+2$ vector

$$\mathbf{v}_i = [x_i\ y_i\ \beta \mathbf{f}_1^D(i)\ \beta \mathbf{f}_2^D(i) \ldots \beta \mathbf{f}_N^D(i)]^{\mathrm{T}}$$

β is a tunable constant

- Build a graph G with M vertices, each pixel location $\mathbf{s}_i \in \Gamma$ have a vertex V_i

Graph Construction (III)

- Weight between vertices V_i and V_j

degree before normalization

$$\rho_i = \sum_{j=1}^{m} \psi(d_{ij})$$

$$w_{ij} = (\rho_i \rho_j)^{-\gamma} \psi(d_{ij})$$

normalization factor γ

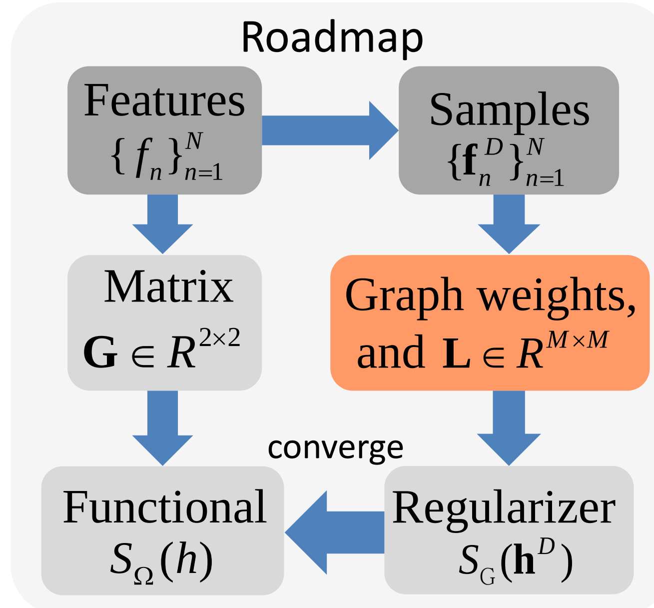
clipped Gaussian kernel

$$\psi(d) = \begin{cases} \exp\left(-\dfrac{d^2}{2\varepsilon^2}\right) & |d| \le r, \\ 0 & \text{otherwise} \end{cases}$$

where $r = \varepsilon C_r$ and C_r is a constant

distance $d_{ij}^2 = \left\| \mathbf{v}_i - \mathbf{v}_j \right\|_2^2$

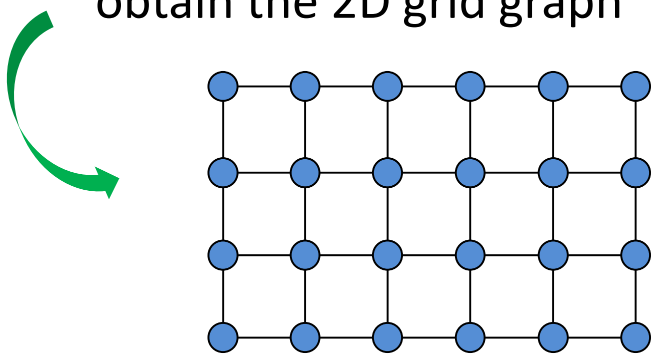
$$= \left\| \mathbf{s}_i - \mathbf{s}_j \right\|_2^2 + \beta^2 \sum_{n=1}^{N} (\mathbf{f}_n^D(i) - \mathbf{f}_n^D(j))^2$$

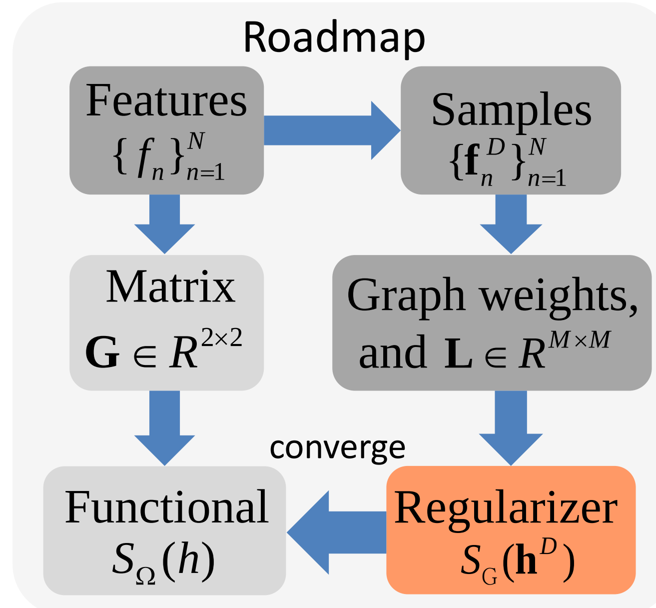


- G is an *r*-neighborhood graph, *i.e.*, no edge connecting two vertices with distance greater than r

Graph Construction (IV)

- Our graph G is very general
  - *e.g.*, choose a small β with proper r, obtain the 2D grid graph

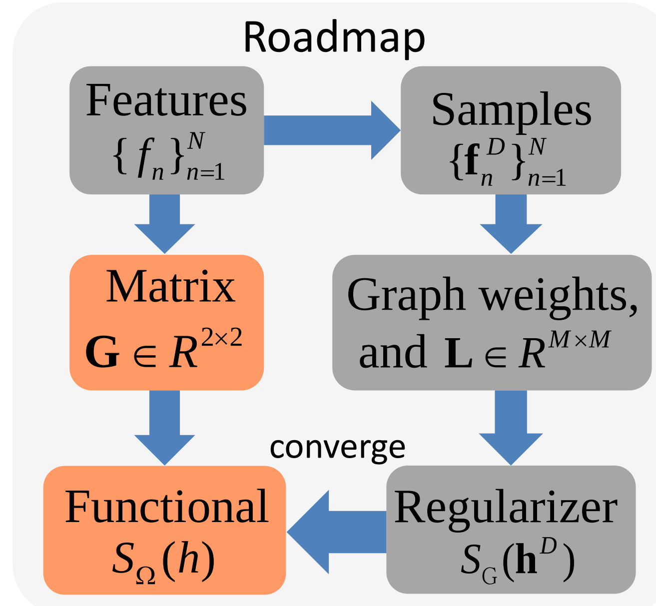




- $\mathbf{A}$ — its (i, j) entry is w_{ij}; $\mathbf{D}$ — its (i, j) entry is $\sum_{j=1}^{m} w_{ij}$ } unnormalized Graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$
- $h(x, y): \Omega \to R$ is a continuous **candidate function**

  $\mathbf{h}^D = [h(x_1, y_1)\; h(x_2, y_2) \ldots h(x_M, y_M)]^\mathrm{T}$ — samples of $h(x, y)$
- $S_\mathrm{G}(\mathbf{h}^D) = (\mathbf{h}^D)^\mathrm{T} \mathbf{L} \mathbf{h}^D$ — graph Laplacian regularizer, a functional on R^M

Convergence of the Graph Laplacian Regularizer (I)



- The continuous counterpart of S_G is a functional S_Ω on domain Ω

$$S_\Omega(h) = \iint_\Omega (\nabla h)^{\mathrm{T}} \mathbf{G}^{-1} (\nabla h) \left(\sqrt{\det \mathbf{G}}\right)^{2\gamma-1} dxdy$$

$\nabla h = [\partial_x h \;\; \partial_y h]^{\mathrm{T}}$ is the gradient of h

- $\mathbf{G}$ is a 2-by-2 matrix:

$$\mathbf{G} = \mathbf{I} + \beta^2 \begin{bmatrix} \sum_{n=1}^N (\partial_x f_n)^2 & \sum_{n=1}^N \partial_x f_n \cdot \partial_y f_n \\ \sum_{n=1}^N \partial_x f_n \cdot \partial_y f_n & \sum_{n=1}^N (\partial_y f_n)^2 \end{bmatrix} = \mathbf{I} + \beta^2 \sum_{n=1}^N \nabla f_n \cdot (\nabla f_n)^{\mathrm{T}}$$

2x2 identity matrix

Structure tensor [3] of the gradients $\{\nabla f_n(x,y)\}_{n=1}^N$

- $\mathbf{G}$ is computed from $\{\nabla f_n\}_{n=1}^N$ on a point-by-point basis

[3] H. Knutsson, C.-F. Westin, and M. Andersson, "**Representing local structure using tensors ii**," in *Image Analysis*. Springer, 2011, vol. 6688, pp. 545–556.

Convergence of the Graph Laplacian Regularizer (II)

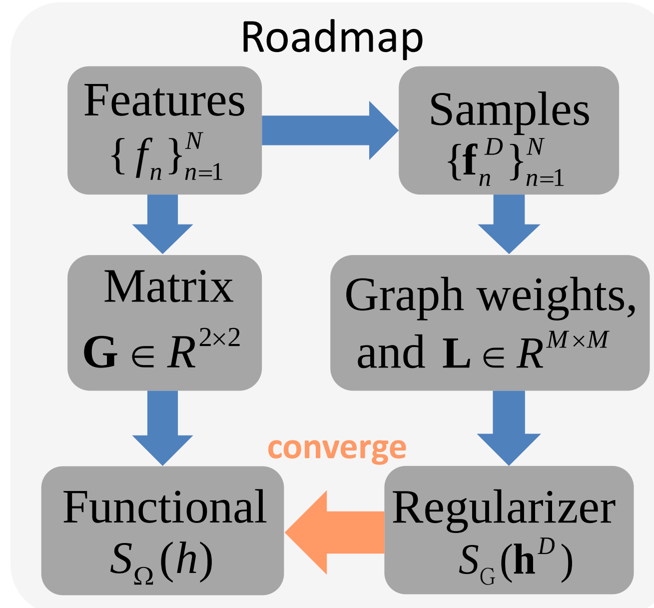
- ***Theorem :** convergence of* S_G *to* S_Ω

$$\lim_{\substack{M\to\infty \\ \varepsilon\to 0}} \frac{M^{2\gamma-1}}{\varepsilon^{4(1-\gamma)}(M-1)} S_G\left(\mathrm{h}^D\right) \sim S_\Omega(h)$$

number of samples M increases

neighborhood $r = \varepsilon C_r$ shrinks

"~" means there exist a constant such that equality holds.



- With results of [4], we proved it by viewing a graph as proxy of an $N+2$-dimensional Riemannian manifold

Vertex	Coordinate on **Ω**	Coordinate on (N+2)-D manifold
V_i	$\mathbf{s}_i = (x_i, y_i)$	$\mathbf{v}_i = [x_i \; y_i \; \beta\mathbf{f}_1^D(i) \; \beta\mathbf{f}_2^D(i) \ldots \beta\mathbf{f}_N^D(i)]^T$

[4] M. Hein, "**Uniform convergence of adaptive graph-based regularization**," in *Learning Theory*. Springer, 2006, pp. 50–64.

Outline

- Introduction
- Convergence of the
  Graph Laplacian Regularizer
- Justification of the
  Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal
  Graph Laplacian Regularizer
- Conclusion

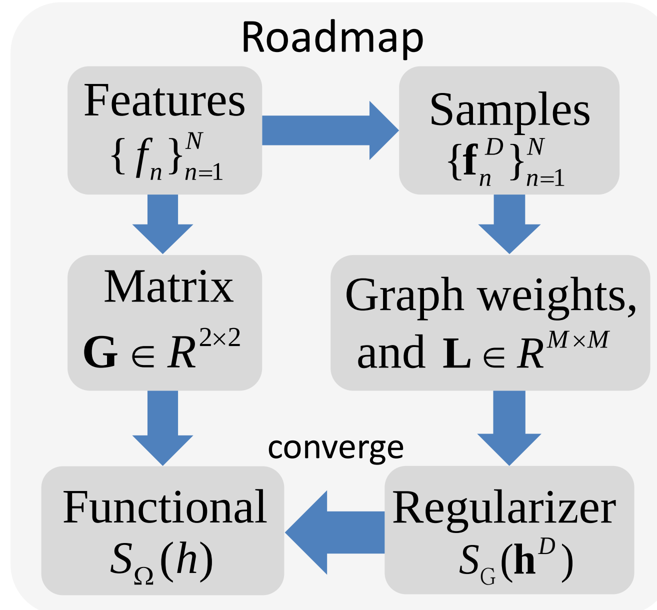
Lena, $\sigma = 30$

Justification of Graph Laplacian Regularizer (I)

$$S_\Omega(h) = \iint_\Omega (\nabla h)^{\mathrm{T}} \mathbf{G}^{-1} (\nabla h) \left(\sqrt{\det \mathbf{G}}\right)^{2\gamma-1} dxdy$$

$$\mathbf{G} = \mathbf{I} + \beta^2 \sum_{n=1}^{N} \nabla f_n \cdot (\nabla f_n)^{\mathrm{T}}$$

$$S_{\mathrm{G}}(\mathbf{h}^D) = (\mathbf{h}^D)^{\mathrm{T}} \mathbf{L} \mathbf{h}^D$$

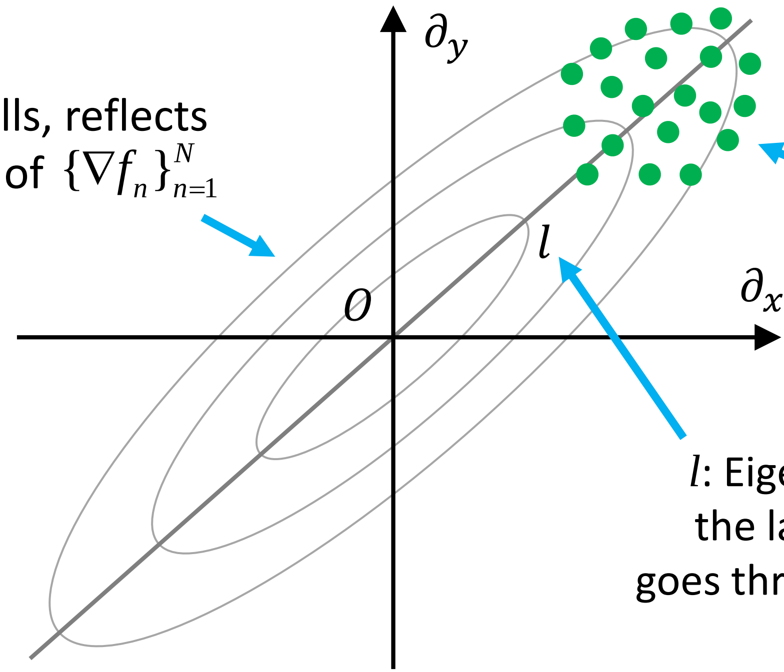


- S_{G} converges to S_Ω, With S_Ω, any new *insights* we can gain on S_{G} ??
    - The eigen-space of $\mathbf{G}$ reflects statistics of $\{\nabla f_n\}_{n=1}^N$
    - $(\nabla h)^{\mathrm{T}} \mathbf{G}^{-1} (\nabla h)$ measures length of ∇h in a metric space established by $\mathbf{G}$!
    - S_Ω integrates the gradient norm

Justification of Graph Laplacian Regularizer (II)

- Metric space defiend by $\mathbf{G}$

Ellipses are norm-balls, reflects how concentration of $\{\nabla f_n\}_{n=1}^{N}$



Green dots are $\{\nabla f_n(x, y)\}_{n=1}^{N}$

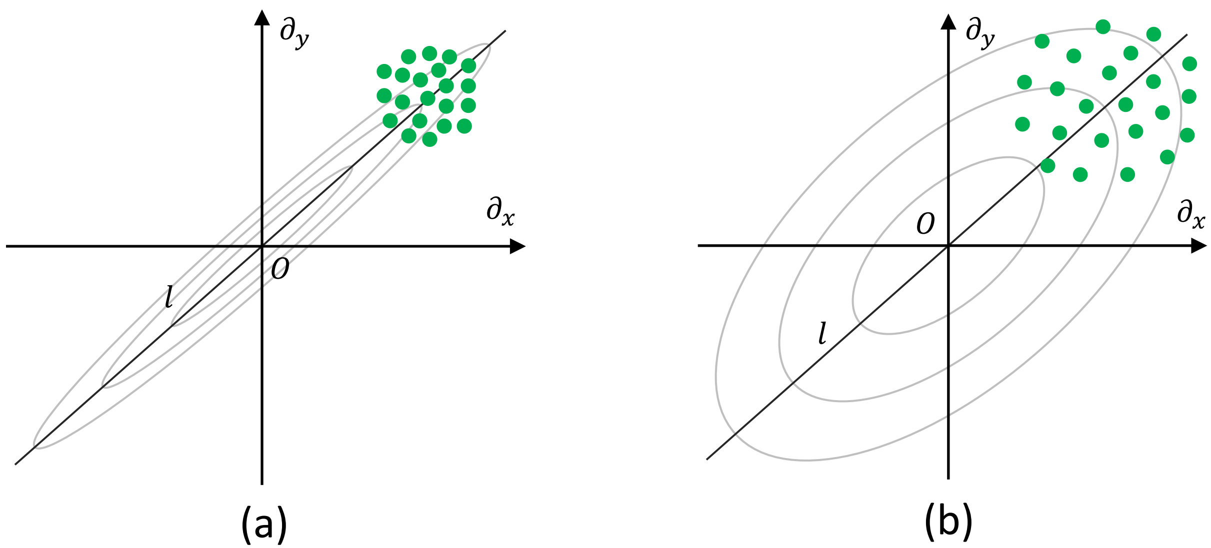
l: Eigenvector corresponds to the largest eigenvalue of $\mathbf{G}$, goes through the cluster of $\{\nabla f_n\}_{n=1}^{N}$

$$S_\Omega(h) = \iint_\Omega (\nabla h)^{\mathrm{T}} \mathbf{G}^{-1} (\nabla h) \left(\sqrt{\det \mathbf{G}}\right)^{2\gamma-1} dxdy$$

$$\mathbf{G} = \mathbf{I} + \beta^2 \sum_{n=1}^{N} \nabla f_n \cdot \left(\nabla f_n\right)^{\mathrm{T}}$$

Justification of Graph Laplacian Regularizer (III)

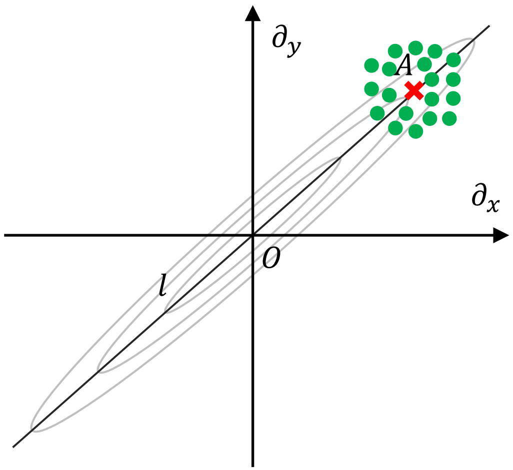
- *The 2D metric space provides a clear picture of what signals are being discriminated and to what extent, on a point-by-point basis in the continuous domain!*



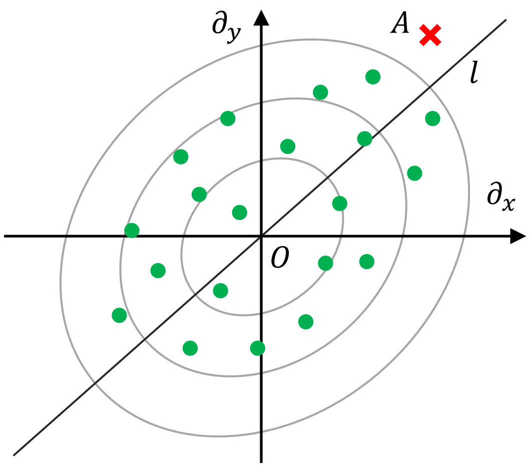
- (a) is more skewed, or discriminant, than (b)
- In (a), a small distance away from the direction orthogonal to l brings large metric distance

Justification of Graph Laplacian Regularizer (IV)

- **Lesson**: Select feature functions properly!
- Suppose A is the truth gradient, choose $\{f_n\}_{n=1}^{N}$ such that
  - (i) l goes through A; (ii) Ellipses stretched flat along l.



(a) A good scheme, $\{\nabla f_n\}_{n=1}^{N}$ are similar to the ground-truth A



(b) A bad scheme...

- For the case of discrete images, one can seek for similar patches in terms of gradient!

Outline

- Introduction
- Convergence of the Graph Laplacian Regularizer
- Justification of the Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal Graph Laplacian Regularizer
- Conclusion

Lena, $\sigma = 30$

Problem Formulation and Algorithm Development

- Adopt a patch-based recovery framework to denoise the image
- For a noisy patch $\mathbf{p}_0$ on the image
  1. Assume a "**self-similar-in-gradient**" image model, search for $K-1$ patches similar to $\mathbf{p}_0$ in terms of gradient in *pre-filtered* image.
  2. Compute graph Laplacian from the similar patches.
  3. Solve the unconstrained quadratic optimization iteratively:

$$\mathbf{q}^{\cdot} = \arg\min_{\mathbf{q}} \ \|\mathbf{p}_0 - \mathbf{q}\|_2^2 + \lambda \mathbf{q}^{\mathrm{T}} \mathbf{L} \mathbf{q}$$

  to obtain the denoised patch $\mathbf{q}^{\cdot}$

- Aggregate denoised patches to form an updated image.
- Denoise the given image iteratively to gradually enhance its quality.
- Our denoising method is named

  Graph-based Denoising using Gradient-based Self-similarity (GDGS)

Outline

- Introduction
- Convergence of the Graph Laplacian Regularizer
- Justification of the Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal Graph Laplacian Regularizer
- Conclusion

Lena, $\sigma = 30$

Experimental Results (I)

- Test images: *Lena*, *Barbara*, *Boats* and *Peppers*
- i.i.d. Additive White Gaussian Noise (AWGN)
- Non-Local GBT (NLGBT) – an existing graph-based denoising method [5]
- Compared to BF, NLM and NLGBT

1.4 dB better than NLM!

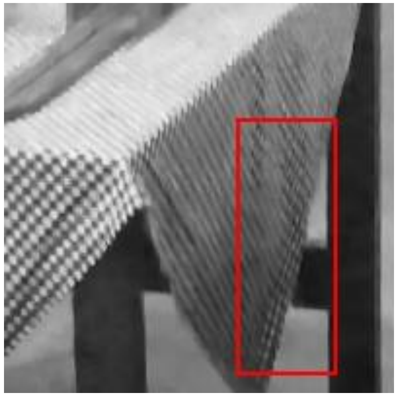
Performance comparisons in PSNR (dB)

Image	Method		Standard Deviation					
			15		20		30	
Lena	GDGS	NLM	**33.47**	32.03	**32.35**	31.51	**30.61**	29.45
	BF	NLGBT	27.00	33.22	24.80	31.90	21.52	30.19
Barbara	GDGS	NLM	**31.71**	30.76	**30.33**	30.15	**28.33**	27.91
	BF	NLGBT	25.78	31.22	23.86	29.62	21.03	27.67
Boats	GDGS	NLM	**31.59**	30.69	**30.30**	29.74	**28.55**	27.68
	BF	NLGBT	26.42	31.05	24.89	29.56	22.19	27.77
Peppers	GDGS	NLM	**33.30**	31.96	**32.38**	31.48	**30.83**	29.50
	BF	NLGBT	28.96	33.18	24.67	32.09	21.49	30.49

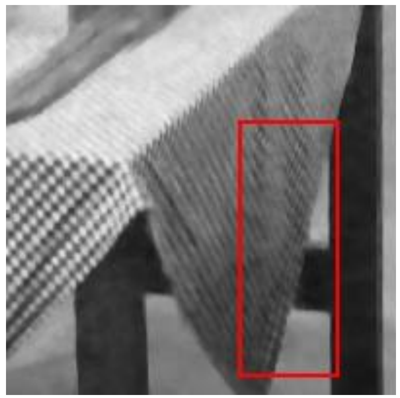
[5] W. Hu, X. Li, G. Cheung, and O. Au, "**Depth map denoising using graph-based transform and group sparsity**," in *IEEE Int'l Workshop on Multimedia Signal Processing*, 2013.

Experimental Results (II)

- GDGS vs NLGBT



GDGS



NLGBT

GDGS

NLGBT

- GDGS vs NLM

GDGS (31.39 dB)

NLM (30.38 dB)

GDGS (29.34 dB)

NLM (28.62 dB)

Noise standard deviation $\sigma = 25$

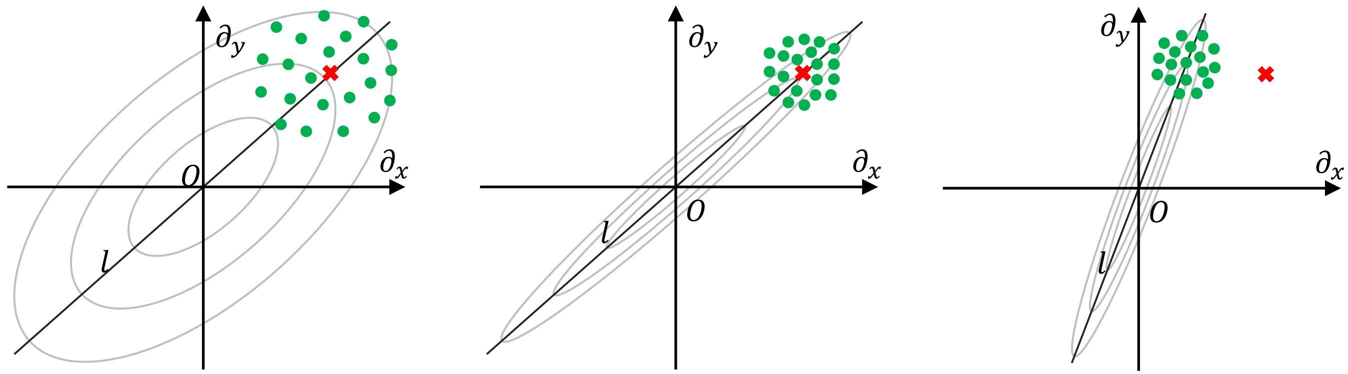
Outline

- Introduction
- Convergence of the Graph Laplacian Regularizer
- Justification of the Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal Graph Laplacian Regularizer
- Conclusion

Lena, $\sigma = 30$

Towards Optimal Graph Laplacian Regularization

- Our latest work [6] derives the optimal metric space $\mathbf{G}^\cdot$, leading to optimal graph Laplacian regularization for denoising.



- *Metric space should be discriminant to the extent that estimates of ground-truth gradient are reliable.*

posterior prob. of ground truth

$$\mathbf{G}^\cdot = \arg\min_{\mathbf{G}} \iint_\Delta \left\|\mathbf{G} - \mathbf{G}_0(\mathbf{g})\right\|_F^2 Pr\left(\mathbf{g} \mid \{\mathbf{g}_k\}_{k=0}^{K-1}\right) d\mathbf{g}$$

Δ—whole gradient domain

ideal metric space given ground truth $\mathbf{g}$

[6] Jiahao Pang, Gene Cheung, Antonio Ortega, Oscar C. Au, "**Optimal Graph Laplacian Regularization for Natural Image Denoising**," submitted to *IEEE ICASSP*, Brisbane, Australia, April, 2015.

Outline

- Introduction
- Convergence of the
  Graph Laplacian Regularizer
- Justification of the
  Graph Laplacian Regularizer
- Formulation and Algorithm
- Experimental Results
- Towards the Optimal
  Graph Laplacian Regularizer
- Conclusion

Lena, $\sigma = 30$

Conclusion

- Image denoising is an ill-posed problem and requires good priors for regularization.
- graph Laplacian regularizer with Gaussian kernel weights converges to a continuous functional.
- Analysis of the continuous functional provides theoretical justification of why and under what conditions the graph Laplacian regularizer can be discriminant.
- Our denoising algorithm with graph Laplacian regularizer and gradient-based similarity out-performs NLM by up to 1.4 dB.
- Our latest work obtains the optimal graph Laplacian, which is discriminant when the estimates are accurate, and robust when the estimates are not.

大学共同利用機関法人 情報・システム研究機構
国立情報学研究所
National Institute of Informatics

Thank You!

Contact: Gene Cheung (cheung@nii.ac.jp)

Jiahao Pang (jpang@ust.hk)