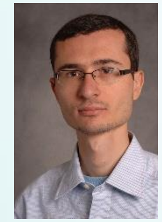
Gene Cheung
National Institute of Informatics
2nd September, 2016

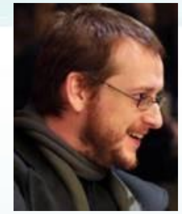
Inverse Imaging Problems using Graph-Signal Smoothness Priors

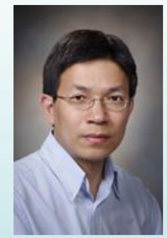
Acknowledgement

Collaborators:

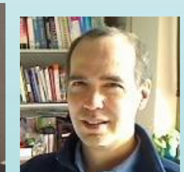
- B. Motz, Y. Mao, Y. Ji (NII, Japan)
- W. Hu, P. Wan, W. Dai, J. Pang, J. Zeng, A. Zheng, O. Au (HKUST, HK)
- Y.-H. Chao, A. Ortega (USC, USA)
- D. Florencio, C. Zhang, P. Chou (MSR, USA)
- Y. Gao, J. Liang (SFU, Canada)
- L. Toni, A. De Abreu, P. Frossard (EPFL, Switzerland)
- C. Yang, V. Stankovic (U of Strathclyde, UK)
- X. Wu (McMaster U, Canada)
- P. Le Callet (U of Nantes, France)
- H. Zheng, L. Fang (USTC, China)
- C.-W. Lin (National Tsing Hua University, Taiwan)
- S. Yang, J. Liu, Z. Guo (Peking U. China)

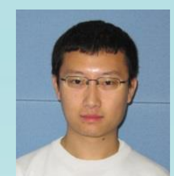






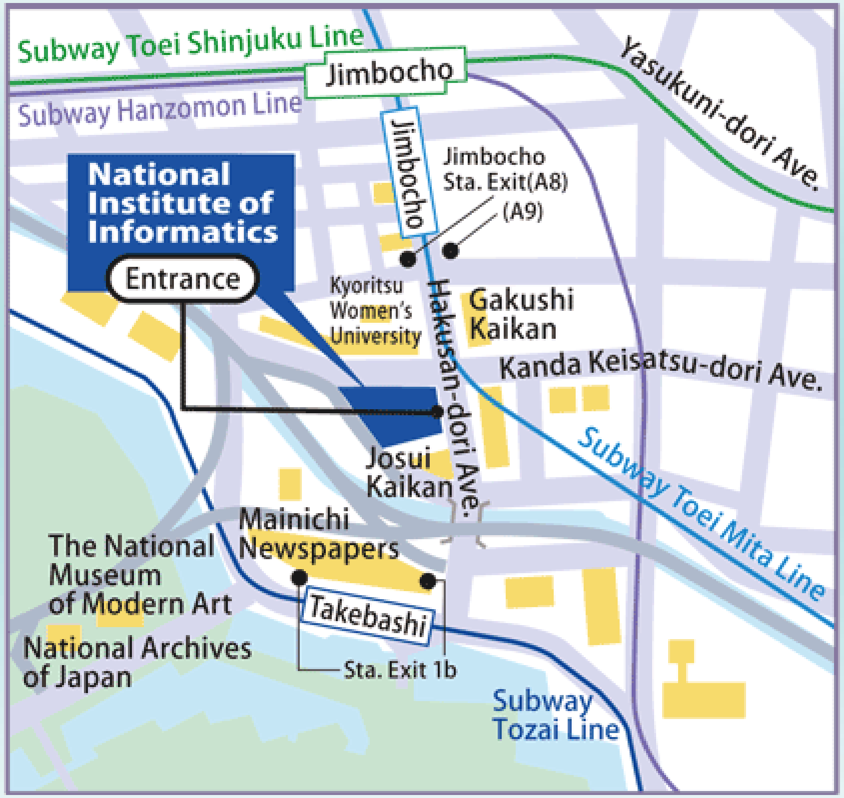




NII Overview

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

- **National Institute of Informatics**
- Chiyoda-ku, Tokyo, Japan.
- Government-funded research lab.

- Offers graduate courses & degrees through **The Graduate University for Advanced Studies** (Sokendai).
- 60+ faculty in "**informatics**": quantum computing, discrete algorithms, database, machine learning, computer vision, speech & audio, image & video processing.



- **Get involved!**
  - 2-6 month Internships.
  - Short-term visits via MOU grant.
  - Lecture series, Sabbatical.

Introduction to APSIPA and APSIPA DL

APSIPA Mission: To promote broad spectrum of research and education activities in signal and information processing in Asia Pacific

APSIPA Conferences: ASPIPA Annual Summit and Conference

APSIPA Publications: Transactions on Signal and Information Processing in partnership with Cambridge Journals since 2012; APSIPA Newsletters

APSIPA Social Network: To link members together and to disseminate valuable information more effectively

APSIPA Distinguished Lectures: An APSIPA educational initiative to reach out to the community

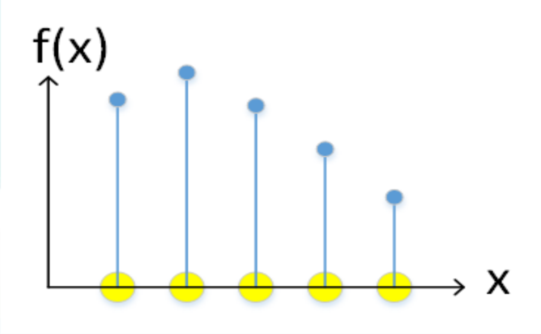
Outline

- Graph Signal Processing
  - Graph spectrum
- Graph-Signal Smoothness Prior
  - Graph Laplacian regularizer
  - LERaG
- Soft Decoding of JPEG Images
  - Mixture of Priors

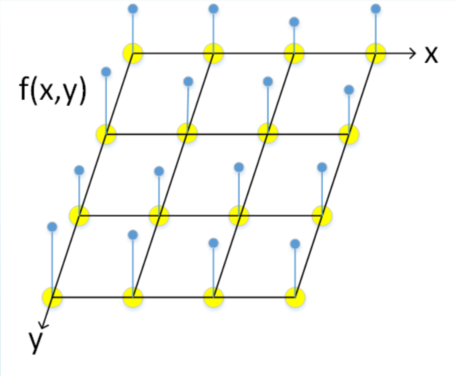
Outline

- Graph Signal Processing
  - Graph spectrum
- Graph-Signal Smoothness Prior
  - Graph Laplacian regularizer
  - LERaG
- Soft Decoding of JPEG Images
  - Mixture of Priors

Digital Signal Processing



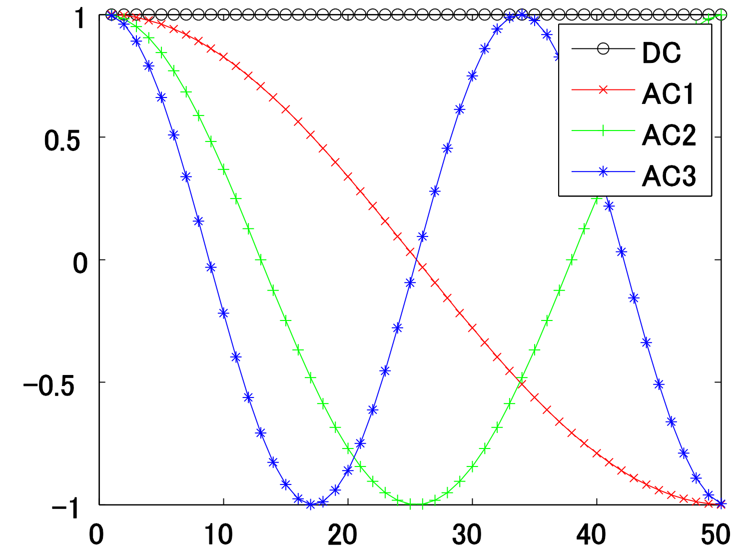
- Discrete signals on ***regular*** data kernels.
  - Ex.1: audio on regularly sampled timeline.
  - Ex.2: image on 2D grid.



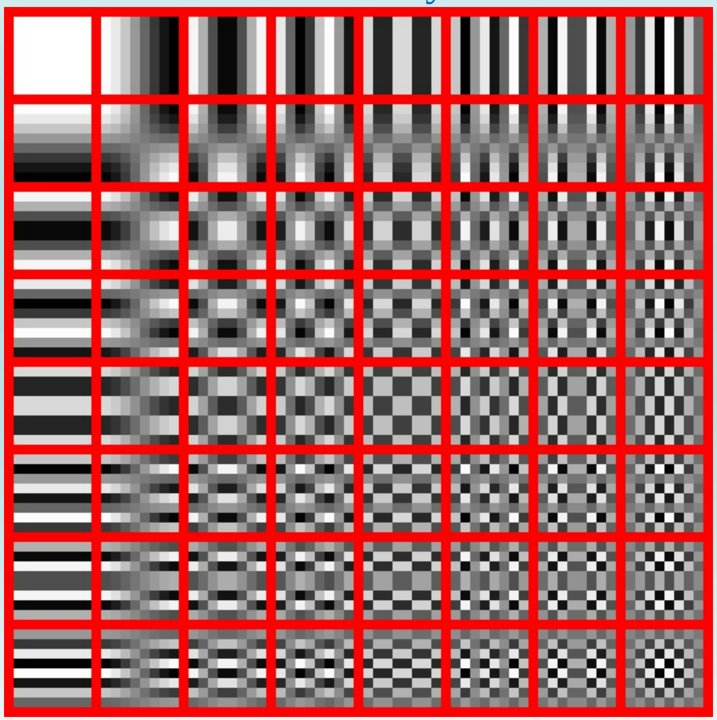
- **Harmonic analysis** tools (transforms, wavelets) for diff. tasks:
  - Compression.
  - Restoration.
  - Segmentation, classification.

Smoothness of Signals

- Signals are often **smooth**.
- Notion of *frequency*, *band-limited*.
- Ex.: **DCT**: $$X_k = \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right)$$



2D DCT basis is set of outer-product of 1D DCT basis in x- and y-dimension.



$$\mathrm{a} = \Phi\, \mathrm{x}$$

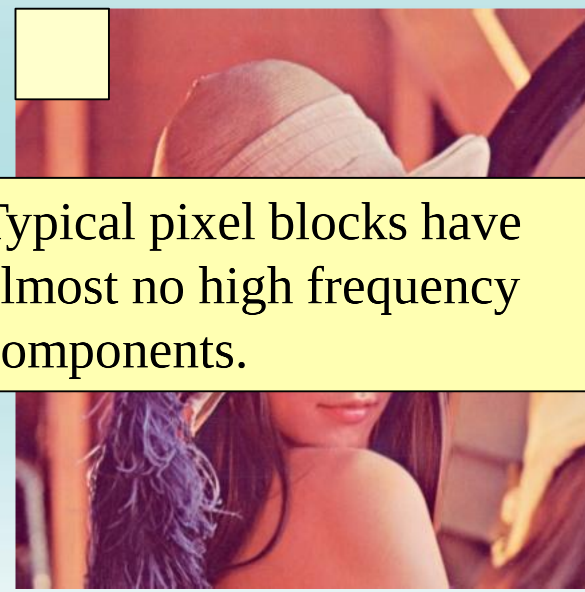
desired signal

transform

transform coeff.

$$\mathrm{a} = \begin{bmatrix} a_0 \\ a_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

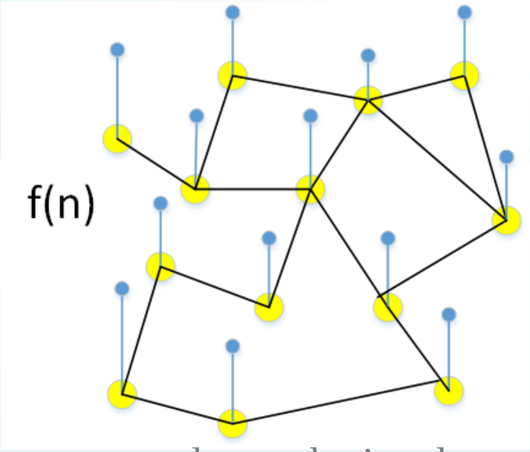
Compact signal representation



Typical pixel blocks have almost no high frequency components.

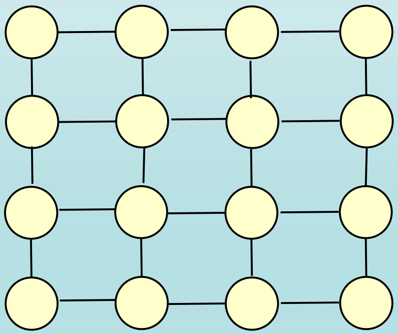
Graph Signal Processing

- Signals on ***structured*** data kernels described by graphs.
  - Graph: nodes and edges.
  - Edges reveals *node-to-node relationships*.



example graph-signal

1. Data domain is naturally a graph.
   - Ex: ages of users on social networks.
2. Underlying data structure unknown.
   - Ex: images: 2D grid → structured graph.



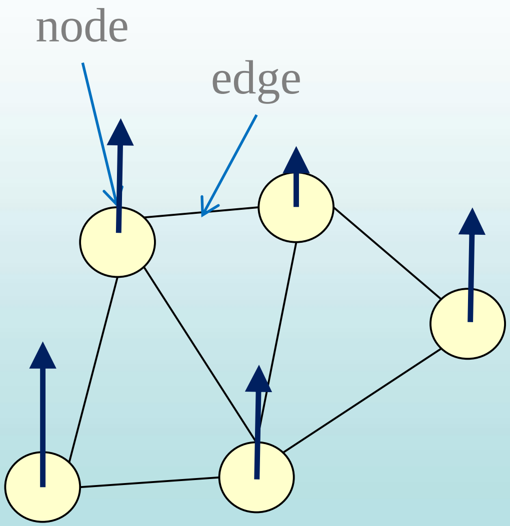
Graph Signal Processing (GSP) addresses the problem of processing signals that live on graphs.

[1] D. I. Shuman et al.,"**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.

Graph Signal Processing

Research questions*:

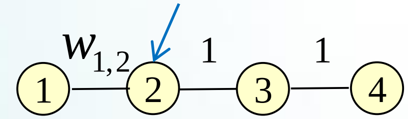
- **Sampling**: how to efficiently acquire / sense a graph-signal?
  - Graph sampling theorems.
- **Representation**: Given graph-signal, how to compactly represent it?
  - Transforms, wavelets, dictionaries.
- **Signal restoration**: Given noisy and/or partial graph-signal, how to recover it?
  - Graph-signal priors.



* **Graph Signal Processing Workshop**, Philadelphia, US, May 25-27, 2016.
https://alliance.seas.upenn.edu/~gsp16/wiki/index.php?n=Main.Program

Graph Fourier Transform (GFT)

undirected graph



Graph Laplacian:

$$A = \begin{bmatrix} 0 & w_{1,2} & 0 & 0 \\ w_{1,2} & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

- *Adjacency Matrix A*: entry $A_{i,j}$ has *non-negative* edge weight $w_{i,j}$ connecting nodes *i* and *j*.

$$D = \begin{bmatrix} w_{1,2} & 0 & 0 & 0 \\ 0 & w_{1,2}+1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- *Degree Matrix D*: diagonal matrix w/ entry $D_{i,i}$ being sum of column entries in row *i* of *A*.

$$D_{i,i} = \sum_j A_{i,j}$$

$$L = \begin{bmatrix} w_{1,2} & -w_{1,2} & 0 & 0 \\ -w_{1,2} & w_{1,2}+1 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

- *Combinatorial Graph Laplacian L*: *L = D-A*
    - ***L*** is *symmetric* (graph undirected).
    - ***L*** is a *high-pass* filter.
    - ***L*** is related to *2nd derivative*.

$$L_{3,:}x = -x_2 + 2x_3 - x_4$$

$$f''(x) = \lim_{h \to 0} \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

*https://en.wikipedia.org/wiki/Second_derivative

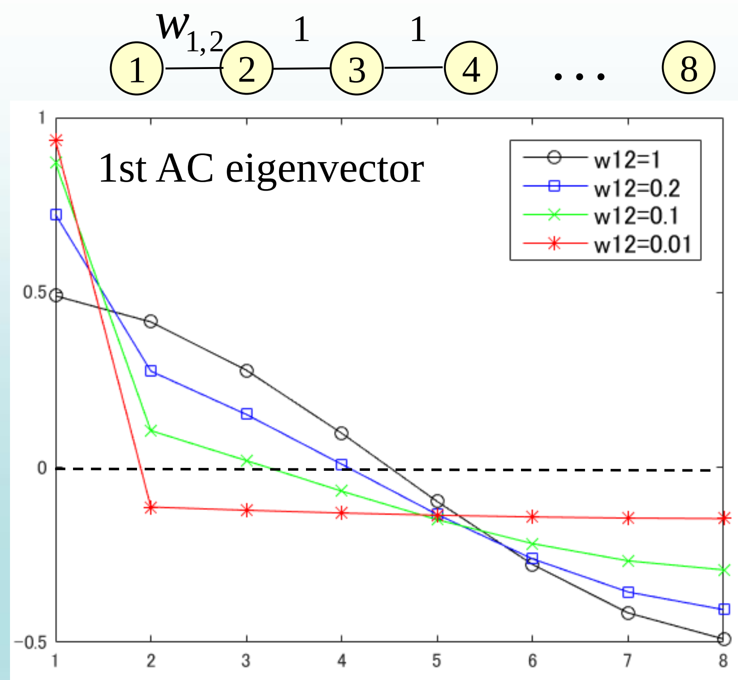
Graph Spectrum from GFT

- **Graph Fourier Transform** (GFT) is eigen-matrix of graph Laplacian *L*.

eigenvalue

$$L\,u_i = \lambda_i\,u_i$$

eigenvector



1. Edge weights affect shapes of eigenvectors.
2. Eigenvalues (≥ 0) as *graph frequencies*.
   - Constant eigenvector is DC.
   - # *zero-crossings* increases as λ increases.

- GFT defaults to *DCT* for un-weighted connected line.
- GFT defaults to *DFT* for un-weighted connected circle.

Variants of Graph Laplacians

- **Graph Fourier Transform** (GFT) is eigen-matrix of graph Laplacian *L*.

eigenvalue

$$L\,u_i = \lambda_i\, u_i$$

eigenvector

- Other definitions of graph Laplacians:
  - **Normalized** graph Laplacian:

$$L_n = D^{-1/2} L D^{-1/2} = I - D^{-1/2} A D^{-1/2}$$

  - **Random walk** graph Laplacian:

$$L_{rw} = D^{-1} L = I - D^{-1} A$$

  - **Generalized** graph Laplacian [1]:

$$L_g = L + D^*$$

<u>Characteristics:</u>

- Normalized.
- Symmetric.
- No DC component.

- Normalized.
- Asymmetric.
- Eigenvectors not orthog.

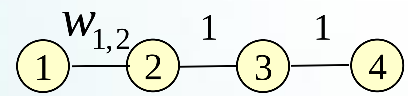
- Symmetric.
- **L** plus self loops.
- Defaults to DST, ADST.

[1] Wei Hu, Gene Cheung, Antonio Ortega, "**Intra-Prediction and Generalized Graph Fourier Transform for Image Coding**," *IEEE Signal Processing Letters*, vol.22, no.11, pp. 1913-1917, November 2015.

Outline

- Graph Signal Processing
  - Graph spectrum
- Graph-Signal Smoothness Prior
  - Graph Laplacian regularizer
  - LERaG
- Soft Decoding of JPEG Images
  - Mixture of Priors

Graph Laplacian Regularizer



- $\mathrm{x}^T \mathrm{L} \mathrm{x}$ (graph Laplacian quadratic form) [1]) is one variation measure → graph-signal smoothness prior.

$$\mathrm{x}^T \mathrm{L} \mathrm{x} = \frac{1}{2} \sum_{i,j} w_{i,j} \left(x_i - x_j \right)^2 = \sum_k \lambda_k \alpha_k^2$$

signal smooth in nodal domain

signal contains mostly low graph freq.

- Signal Denoising:

desired signal

observation → $y = x + v$ ← noise

- MAP formulation:

$$\min_x \left\| \mathrm{y} - \mathrm{x} \right\|_2^2 + \mu \, \mathrm{x}^T \mathrm{L} \mathrm{x}$$

fidelity term

smoothness prior

[1] Jiahao Pang, Gene Cheung, "**Graph Laplacian Regularization for Inverse Imaging: Analysis in the Continuous Domain**," submitted to *IEEE Transactions on Image Processing*, April 2016.

Graph Laplacian Regularizer for Denoising

1. Choose graph:
   - Connect neighborhood graph.
   - Assign edge weight:

pixel intensity difference

pixel location difference

$$w_{i,j} = \exp\left(\frac{-\|x_i - x_j\|_2^2}{\sigma_1^2}\right)\exp\left(\frac{-\|l_i - l_j\|_2^2}{\sigma_2^2}\right)$$

2. Solve obj. in closed form:

$$\min_{x} \|\mathrm{y} - \mathrm{x}\|_2^2 + \mu\, \mathrm{x}^T \mathrm{L} \mathrm{x}$$

- Iterate until convergence.

TABLE I

PERFORMANCE COMPARISON IN PSNR (DB) AT $QF = 15$

Images	$QF = 15$					
	JPEG	TV	DicTV	TGV	ANCE	Prop
Dude	37.00	34.23	37.58	37.19	37.99	**38.17**
Teddy	31.46	32.30	31.60	31.52	32.14	**32.33**
Tsukuba	33.13	35.29	34.19	33.68	34.69	**36.22**
Ballet	35.63	36.48	36.77	36.15	37.28	**37.49**
Champagne	36.82	34.12	37.46	37.00	**37.73**	37.68
Gain	1.57	1.89	0.86	1.27	0.41	-

Comments:

1. **L** is NOT normalized.

2. Why works well for PWS signals?

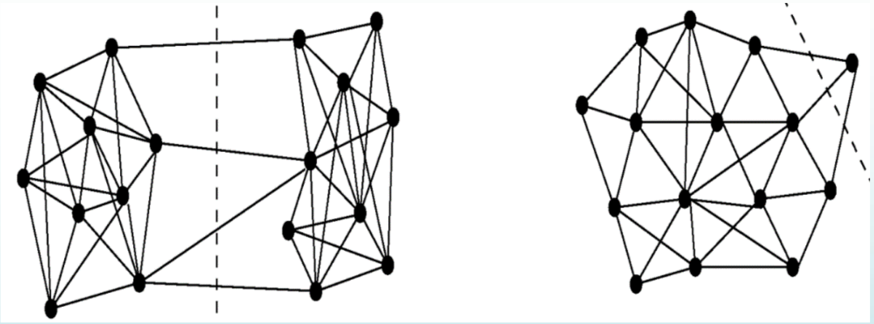
[1] W. Hu, G. Cheung, M. Kazui, "**Graph-based Dequantization of Block-Compressed Piecewise Smooth Images**," *IEEE Signal Processing Letters*, vol.23, no.2, pp.242-246, February 2016.

Spectral Clustering

- **Normalized Cut** [1]:

$$\min_{A,B} Ncut(A,B) := cut(A,B)\left(\frac{1}{vol(A)} + \frac{1}{vol(B)}\right)$$

$$cut(A,B) = \sum_{i\in A,\, j\in B} W_{i,j} \qquad vol(A) = \sum_{i\in A} D_{i,i}$$



min normalized cut　　　min cut

- Problem is **NP-hard**, so:

1. Rewrite as:

$$\min_{\mathrm{f}} \frac{\mathrm{f}^T \mathrm{L} \mathrm{f}}{\mathrm{f}^T \mathrm{D} \mathrm{f}} \quad s.t.\ f_i = \begin{cases} \dfrac{1}{vol(A)} & if\ i \in A \\ \dfrac{-1}{vol(B)} & if\ i \in B \end{cases}$$

2. Relax to:

$$\min_{\mathrm{f}} \frac{\mathrm{f}^T \mathrm{L} \mathrm{f}}{\mathrm{f}^T \mathrm{D} \mathrm{f}} \quad s.t.\ \mathrm{f}^T \mathrm{D} 1 = 0$$

[1] J. Shi and J. Malik, "**Normalized cuts and image segmentation**," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

Eigenvectors of Normalized graph Laplacian

$$\min_{x} \|y - x\|_2^2 + \mu\, x^T L_n x$$

candidate objective

$$\sum_k \tilde{\eta}_k \alpha_k^2$$

- Define:

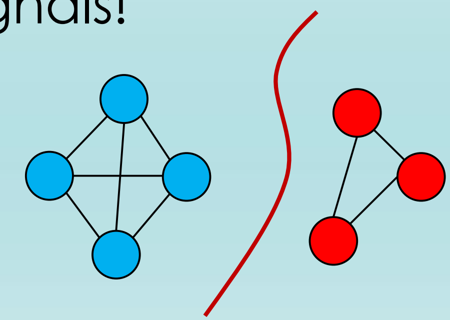
$$v := D^{1/2} f \qquad v_1 := D^{1/2} 1$$

- Problem rewritten as:

Rayleigh quotient

$$v^* = \arg\min_{v} \frac{v^T L_n v}{v^T v} \quad s.t.\ v^T v_1 = 0$$

  - $\mathbf{v}_1$ minimizes obj → Sol'n is 2nd eigenvector of $\mathbf{L}_n$.
  - If **f*** optimal to norm. cut, **v*** is PWS → well rep. PWS signals!
  - **f*** optimal when nodes easy to cluster:
    - Easy-to-cluster graph has small **Fiedler number**.



- **Disadvantage**:
  - $\mathbf{v}_1$ not constant vector (DC) → cannot well rep. smooth patch.

Eigenvectors of random walk graph Laplacian

- **Random walk graph Laplacian $\mathbf{L}_{rw}$** is **similar**:

$$\mathrm{L}_{\mathrm{rw}} = \mathrm{D}^{-1/2}\mathrm{L}_n\mathrm{D}^{1/2}$$

- Let: $\mathrm{L}_\mathrm{n} = \mathrm{V}\Lambda\mathrm{V}^T$ ← eigen-decomposition

- $\mathbf{L}_{rw}$ has **left eigenvectors** $\mathrm{V}^T\mathrm{D}^{1/2}$:

diagonal matrix
w/ eigenvalues

$$\mathrm{V}^T\mathrm{D}^{1/2}\mathrm{L}_{rw} = \mathrm{V}^T\mathrm{D}^{1/2}\mathrm{D}^{-1/2}\mathrm{V}\Lambda\mathrm{V}^T\mathrm{D}^{1/2} = \Lambda\mathrm{V}^T\mathrm{D}^{1/2}$$

- **Advantage**:
  - Constant signal **1** maps to $\mathrm{V}^T\mathrm{D}^{1/2} \rightarrow \mathrm{D}^{1/2}$ maps to $\mathrm{V}^T \rightarrow \mathbf{v}_1$
    - $\beta = \mathrm{V}^T\mathrm{D}^{1/2}1$ is sparse.

[1] Xianming Liu, Gene Cheung, Xiaolin Wu, Debin Zhao, "**Random Walk Graph Laplacian based Smoothness Prior for Soft Decoding of JPEG Images**," submitted to *IEEE Transactions on Image Processing*, July 2016.

Left E-vector random walk graph Laplacian (LERaG)

- **Disadvantage**:
  - $\mathbf{L}_{rw}$ is asymmetric → no orthogonal e-vectors w/ real e-values.
- So, **left Eigenvector Random Walk Graph Laplacian** (LERaG) [1]:

$$\boxed{\mathbf{x}^T \mathcal{L}_r^T \mathcal{L}_r \mathbf{x} = (\mathbf{x}^T \mathbf{D}^{1/2} \mathcal{L}_n) \mathbf{D}^{-1} (\mathcal{L}_n \mathbf{D}^{1/2} \mathbf{x})}$$

$$\gamma = \mathcal{L}_n \mathbf{D}^{1/2} \mathbf{x}$$

projection of signal **x** to $\mathbf{D}^{1/2}$, then $\mathbf{L}_n$

$$\boxed{\mathbf{x}^T \mathcal{L}_r^T \mathcal{L}_r \mathbf{x} = \gamma^T \mathbf{D}^{-1} \gamma}$$

$$\boxed{\frac{\gamma^T \gamma}{d_{\max}} \leq \gamma^T \mathbf{D}^{-1} \gamma \leq \frac{\gamma^T \gamma}{d_{\min}}} \Rightarrow \boxed{(d_{\min}^{-1}) \gamma^T \gamma}$$

[1] Xianming Liu, Gene Cheung, Xiaolin Wu, Debin Zhao, "**Random Walk Graph Laplacian based Smoothness Prior for Soft Decoding of JPEG Images**," submitted to *IEEE Transactions on Image Processing*, July 2016.

Frequency Interpretation of LERaG

- Promote low graph frequencies:

projection of signal **x** to left e-vectors

$$\gamma^T\gamma = \mathbf{x}^T\mathbf{D}^{1/2}\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T\mathbf{D}^{1/2}\mathbf{x}$$
$$= \boldsymbol{\beta}^T\boldsymbol{\Lambda}^2\boldsymbol{\beta} = \sum_k \tilde{\eta}_k^2\beta_k^2.$$

coeff of $\mathbf{L}_{rm}$'s left e-vectors

e-values of $\mathbf{L}_n$, $\mathbf{L}_{rm}$

- Fast implementation of LERaG:

$$(d_{\min}^{-1})\gamma^T\gamma = \mathbf{x}^T(d_{\min}^{-1})\mathbf{L}\mathbf{D}^{-1}\mathbf{L}\mathbf{x}$$

inverse diagonal matrix

combinatorial Laplacian

Comparison of Graph-signal Smoothness Priors

- Different graph Laplacian matrices
  - Combinatorial graph Laplacian: $\mathbf{L} = \mathbf{D} - \mathbf{W}$
  - Symmetrically normalized graph Laplacian: $\mathcal{L}_n = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$
  - Random walk graph Laplacian: $\mathcal{L}_r = \mathbf{D}^{-1}\mathbf{L}$
  - Doubly stochastic graph Laplacian [1]: $\mathcal{L}_d = \mathbf{I} - \mathbf{C}^{-1/2}\mathbf{W}\mathbf{C}^{-1/2}$

Graph Laplacian	Symmetric	Normalized	DC e-vector
Combinatorial	Yes	No	Yes
Symmetrically Normalized	Yes	Yes	No
Random Walk	No	Yes	Yes
Doubly Stochastic [1]	Yes	Yes	Yes

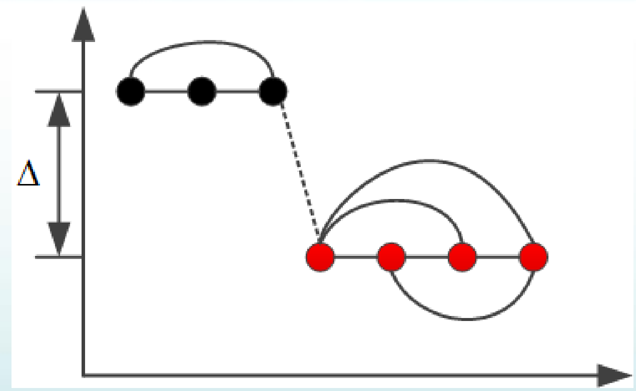
[1] A. Kheradmand and P. Milanfar, "**A general framework for regularized, similarity-based image restoration**," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5136–5151, Dec 2014.

LERaG for PWC Signal

$$w_{i,j} = \exp\left(\frac{-\|x_i - x_j\|_2^2}{\sigma_1^2}\right)\exp\left(\frac{-\|l_i - l_j\|_2^2}{\sigma_2^2}\right)$$

- Construct fully connected graph for PWC signal:

$$\mathbf{W} = \begin{bmatrix} \mathbf{A}_l & \mathbf{0}_{l\times(n-l)} \\ \mathbf{0}_{(n-l)\times l} & \mathbf{A}_{n-l} \end{bmatrix} \quad \mathcal{L}_n = \begin{bmatrix} \tilde{\mathbf{B}}_l & \mathbf{0}_{l\times(n-l)} \\ \mathbf{0}_{(n-l)\times l} & \tilde{\mathbf{B}}_{n-l} \end{bmatrix}$$



PWC signal: 2 constant pieces w/ large gap Δ

- 1st eigenvector for e-value = 0: $\mathbf{v}_1 = \mathbf{D}^{1/2}\mathbf{1}$
- 2nd eigenvector for e-value = 0:

$$v_{2,i} = \begin{cases} 1/l(l-1)^{1/2} & \text{if } 1 \le i \le l \\ -1/(n-l)(n-l-1)^{1/2} & \text{if } l < i \le n \end{cases}$$

- Can be shown that $\mathbf{D}^{1/2}\mathbf{x} = a_1\mathbf{v}_1 + a_2\mathbf{v}_2$
- **LERaG evaluates to 0!** $d_{\min}^{-1}\gamma^T\gamma = \sum_k \tilde{\eta}_k^2 \beta_k^2 = 0$

Outline

- Graph Signal Processing
  - Graph spectrum
- Graph-Signal Smoothness Prior
  - Graph Laplacian regularizer
  - LERaG
- Soft Decoding of JPEG Images
  - Mixture of Priors

Soft Decoding of JPEG Images

- **Setting**: **JPEG** compresses natural images:
  1. Divide image into 8x8 blocks, DCT.
  2. Perform DCT transform per block and quantize:

$$q_i = round(\mathrm{Y}_i / \mathrm{Q}_i), \quad \mathbf{Y} = \mathbf{T}\mathbf{y}$$

DCT

8x8 pixel block

quantization parameter

DCT Coefficients

  3. Quantized DCT coeff entropy coded.

- **Decoder**: uncertainty in signal reconstruction:

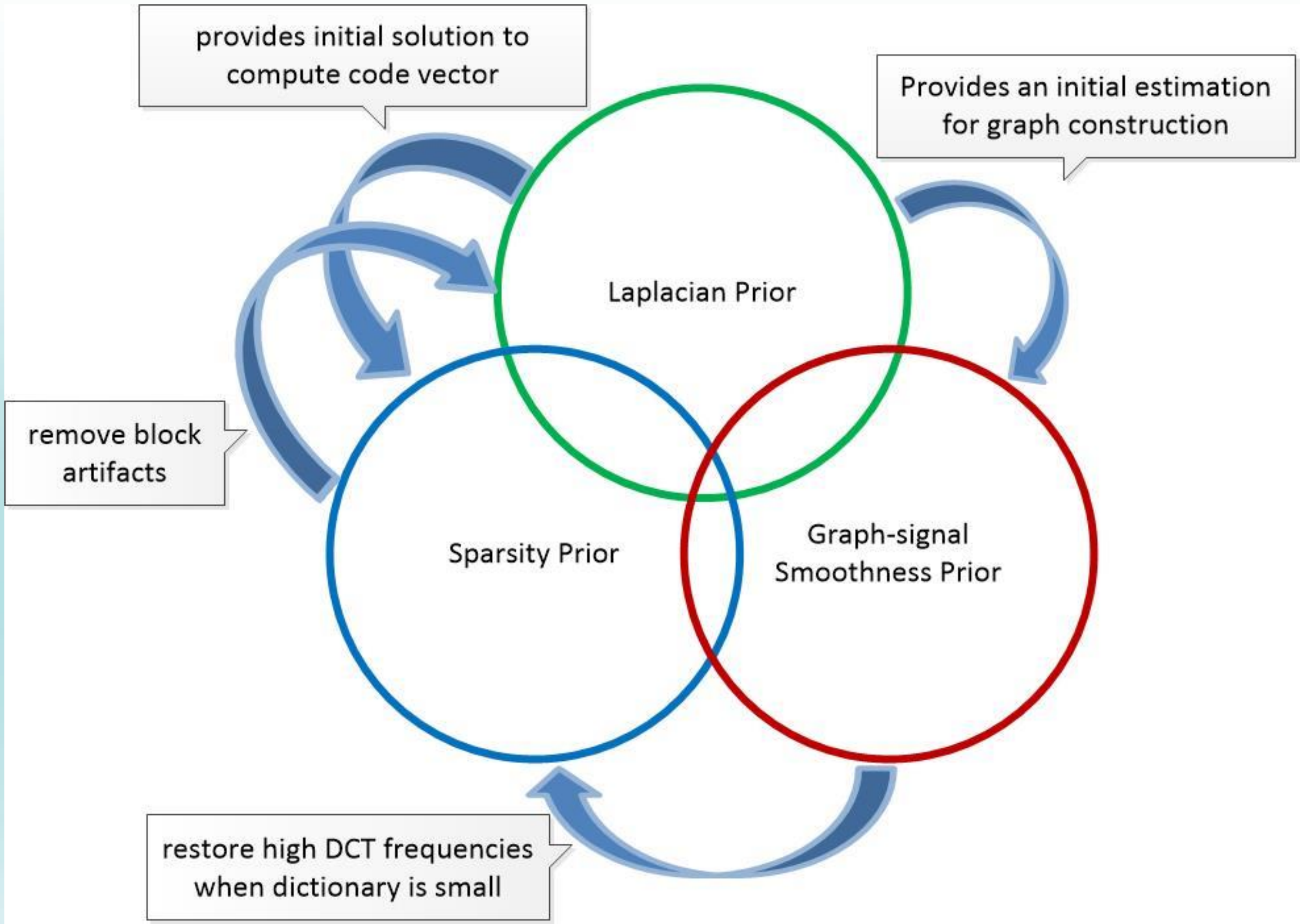
$$q_i \, \mathrm{Q}_i \leq \mathrm{Y}_i \leq (q_i + 1)\, \mathrm{Q}_i, i = 1, 2, \cdots, 64.$$

LERaG for Soft Decoding of JPEG Images

- **Problem**: reconstruct image given indexed quant. bin in 8x8 DCT.

- **Procedure**:

1. Initialize *per-block* MMSE sol'n via Laplacian prior.
2. Solve *per-patch* signal restoration problem w/ 2 priors:
   1. Sparsity prior
   2. Graph-signal smoothness prior



Soft Decoding Algorithm w/ Prior Mixture

- Objective:

fidelity term

sparsity prior

graph-signal smoothness prior

graph-signal, code vector

$$\arg\min_{\{\mathbf{x},\boldsymbol{\alpha}\}} \|\mathbf{x}-\boldsymbol{\Phi}\boldsymbol{\alpha}\|_2^2 + \lambda_1\|\boldsymbol{\alpha}\|_0 + \lambda_2\mathbf{x}^T(d_{\min}^{-1})\mathbf{L}\mathbf{D}^{-1}\mathbf{L}\mathbf{x},$$

$$\text{s.t. } \mathbf{q}\mathbf{Q} \preceq \mathbf{T}\mathbf{M}\mathbf{x} \prec (\mathbf{q}+1)\mathbf{Q}$$

quantization bin constraint

- Optimization:
  1. Laplacian prior provides an initial estimation;
  2. Fix **x** and solve for **a**;
  3. Fix **a** and solve for **x**.

Experimental Setup

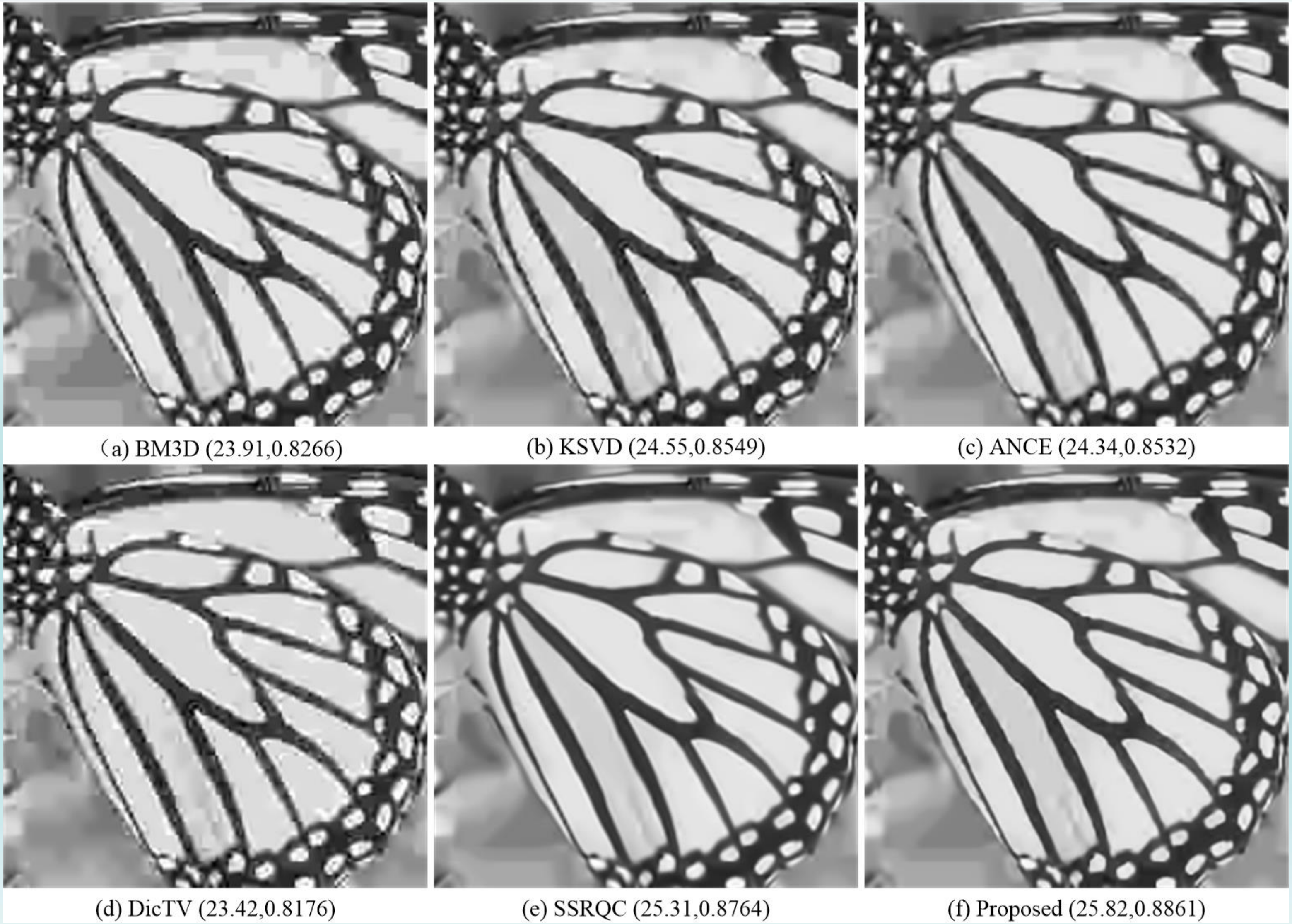
- Compared methods
  - **BM3D**: well-known denoising algorithm
  - **KSVD**: with a large enough over-complete dictionary (100x4000); our method uses a much smaller one (100x400).
  - **ANCE**: non-local self similarity **[Zhang et al. TIP14]**
  - **DicTV**: Sparsity + TV **[Chang et al, TSP15]**
  - **SSRQC**: Low rank + Quantization constraint **[Zhao et al. TCSVT16]**

PSNR / SSIM Comparison

QUALITY COMPARISON WITH RESPECT TO PSNR (IN DB) AND SSIM AT QF = 40

Images	JPEG		BM3D [38]		KSVD [8]		ANCE [18]		DicTV [3]		SSRQC [20]		Ours	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Butterfly	29.97	0.9244	31.35	0.9555	31.57	0.9519	31.38	0.9548	31.22	0.9503	32.02	0.9619	**32.87**	**0.9627**
Leaves	30.67	0.9438	32.55	0.9749	33.04	0.9735	32.74	0.9728	32.45	0.9710	32.13	0.9741	**34.42**	**0.9803**
Hat	32.78	0.9022	33.89	0.9221	33.62	0.9149	33.69	0.9169	33.20	0.8988	34.10	0.9237	**34.46**	**0.9268**
Boat	33.42	0.9195	34.77	0.9406	34.28	0.9301	34.64	0.9362	26.08	0.7550	33.88	0.9306	**34.98**	**0.9402**
Bike	28.98	0.9131	29.96	0.9356	30.19	0.9323	30.31	0.9357	29.75	0.9154	30.35	0.9411	**31.14**	**0.9439**
House	35.07	0.8981	36.09	0.9013	36.05	0.9055	36.12	0.9048	35.17	0.8922	36.49	0.9072	**36.55**	**0.9071**
Flower	31.62	0.9112	32.81	0.9357	32.63	0.9271	32.67	0.9314	31.86	0.9084	33.02	0.9362	**33.37**	**0.9371**
Parrot	34.03	0.9291	34.92	0.9397	34.91	0.9371	35.02	0.9397	33.92	0.9227	35.11	0.9401	**35.32**	**0.9401**
Pepper512	34.21	0.8711	34.94	0.8767	34.89	0.8784	34.99	0.8803	34.24	0.8639	35.05	0.8795	**35.19**	**0.8811**
Fishboat512	32.76	0.8763	33.61	0.8868	33.36	0.8809	33.60	0.8861	32.53	0.8496	33.68	0.8859	**33.73**	**0.8871**
Lena512	35.12	0.9089	36.03	0.9171	35.82	0.9146	36.04	0.9177	34.85	0.8986	36.09	0.9187	**36.11**	**0.9191**
Airplane512	33.36	0.9253	34.38	0.9361	34.36	0.9341	34.53	0.9358	33.75	0.9134	35.81	0.9355	**36.07**	**0.9439**
Bike512	29.43	0.9069	30.47	0.9299	30.66	0.9258	30.71	0.9298	30.05	0.9043	32.26	0.9372	**32.55**	**0.9387**
Statue512	32.78	0.9067	33.61	0.9188	33.55	0.9149	33.55	0.9193	32.53	0.8806	34.88	0.9249	**34.95**	**0.9273**
Average	32.44	0.9097	33.52	0.9264	33.50	0.9229	33.57	0.9258	32.25	0.8945	33.91	0.9283	**34.41**	**0.9311**

Subjective Quality Evaluation



(a) BM3D (23.91,0.8266)
(b) KSVD (24.55,0.8549)
(c) ANCE (24.34,0.8532)
(d) DicTV (23.42,0.8176)
(e) SSRQC (25.31,0.8764)
(f) Proposed (25.82,0.8861)

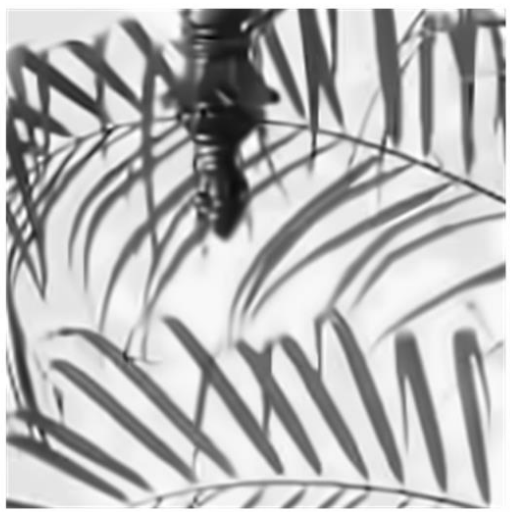
Subjective Quality Evaluation


(a) BM3D (23.78,0.8408)


(b) KSVD (24.39,0.8684)


(c) ANCE (24.18, 0.8551)


(d) DicTV (23.27,0.8245)


(e) SSRQC (25.01,0.8861)


(f) Proposed (25.57,0.8979)

Other Comparisons

- Computation complexity:

TIME	BM3D	KSVD	ANCE	DicTV	SSRQC	Proposed
Average	373.35	209.71	307.43	39.53	70.32	143.73

- Comparisons w/ other graph regularizers:

Images	Combinatorial	Normalized	Doubly Stochastic	LERaG
Butterfly	25.42	24.70	25.15	**25.57**
Leaves	24.99	24.54	24.84	**25.17**
Hat	27.53	27.42	27.43	**27.56**
Boat	26.99	26.94	26.98	**26.99**
Bike	23.12	23.01	23.09	**23.17**
House	29.87	29.83	29.86	**29.89**
Flower	25.84	25.78	25.82	**25.87**
Parrot	27.97	27.95	27.97	**28.02**
Average	26.46	26.27	26.39	**26.53**

Summary

- Graph Signal Processing (GSP)
  - Tools to process signals that live on graphs.
- Graph-signal Smoothness for Inverse Problems
  - New regularizer: **LERaG**
    - Normalized
    - Handles constant / PWC signals
    - Computation-efficient
- Soft Decoding of JPEG Images
  - Mixture of Laplacian, sparsity, graph-signal smoothness prior

Q&A

- Email: cheung@nii.ac.jp
- Homepage: http://research.nii.ac.jp/~cheung/