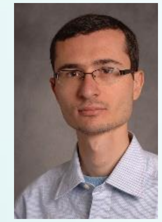
Gene Cheung
National Institute of Informatics
26th November, 2017

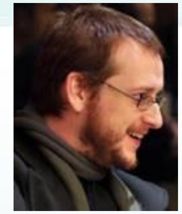
Graph Signal Processing for Image Coding & Restoration

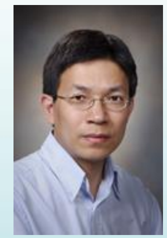
Acknowledgement

Collaborators:

- M. Kaneko (NII, Japan)
- A. Ortega (USC, USA)
- D. Florencio (MSR, USA)
- P. Frossard (EPFL, Switzerland)
- J. Liang, I. Bajic (SFU, Canada)
- V. Stankovic (U of Strathclyde, UK)
- X. Wu (McMaster U, Canada)
- P. Le Callet (U of Nantes, France)
- X. Liu (HIT, China)
- W. Hu, J. Liu, Z. Guo (Peking U., China)
- L. Fang (Tsinghua, HK)
- C.-W. Lin (National Tsing Hua University, Taiwan)

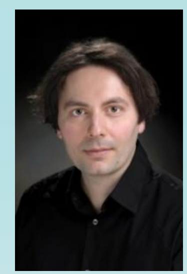


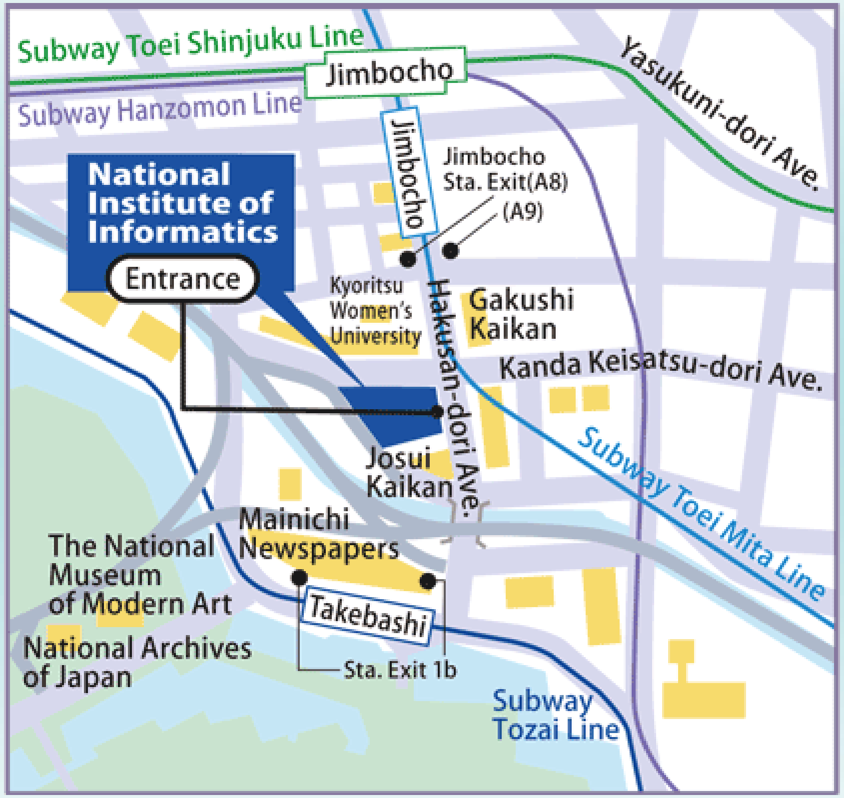






NII Overview

- **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

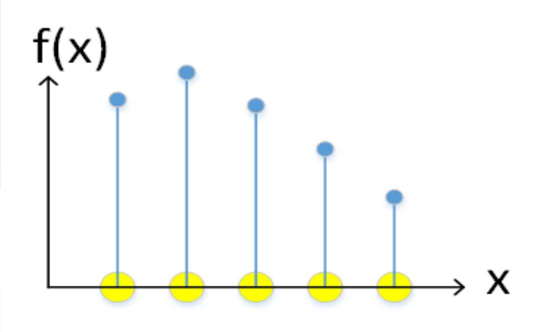
Outline

- Graph Signal Processing
  - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

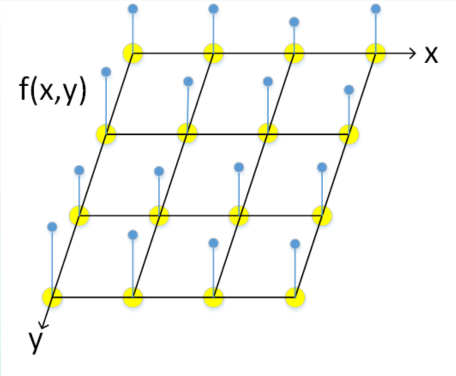
Outline

- Graph Signal Processing
  - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

Digital Signal Processing

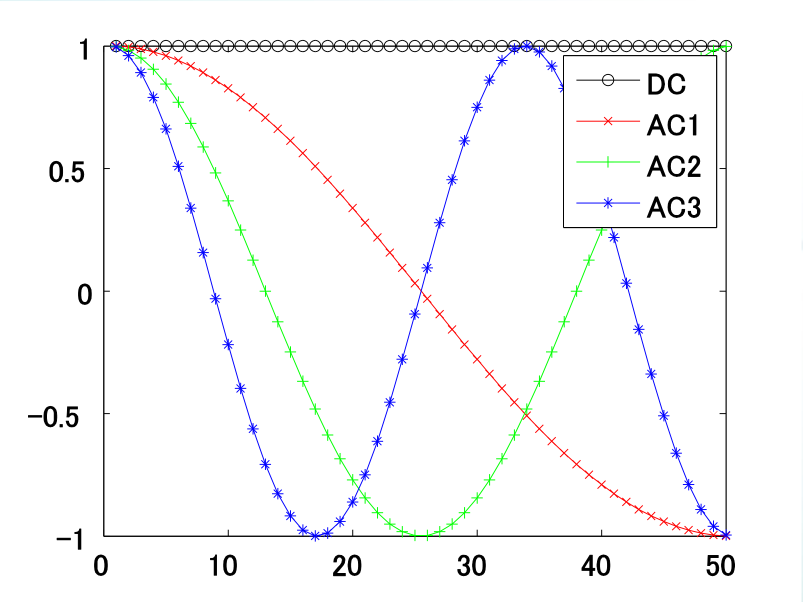


- Discrete signals on ***regular*** data kernels.
  - Ex.1: audio on regularly sampled timeline.
  - Ex.2: image on 2D grid.



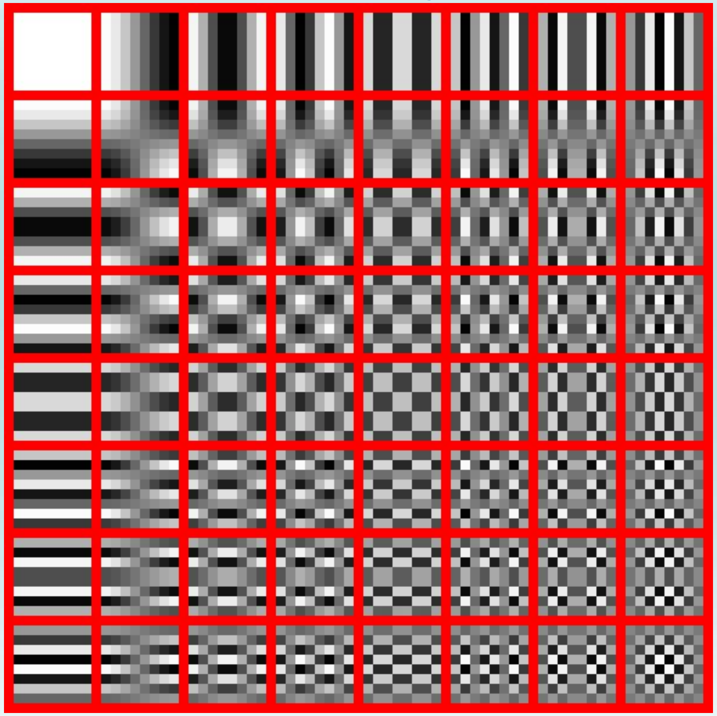
- **Harmonic analysis** tools (transforms, wavelets) for diff. tasks:
  - Compression.
  - Restoration.
  - Segmentation, classification.

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.



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

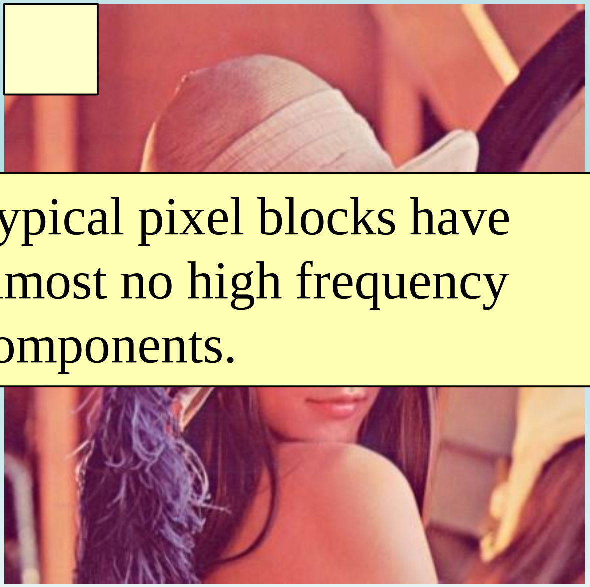
desired signal

transform

transform coeff.

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

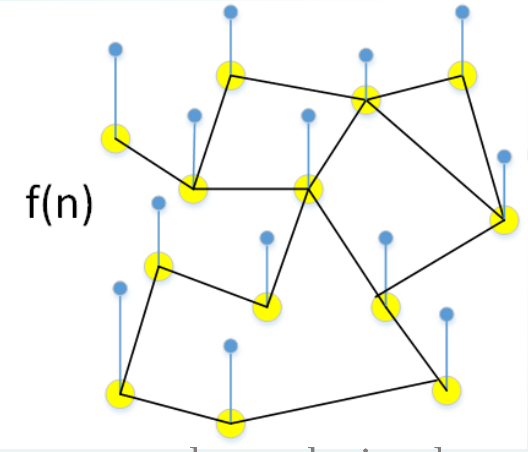
Compact signal representation



Typical pixel blocks have almost no high frequency components.

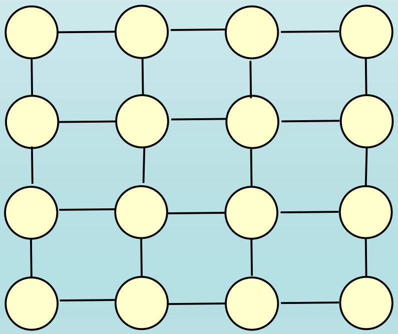
Graph Signal Processing

- Signals on ***irregular*** 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.



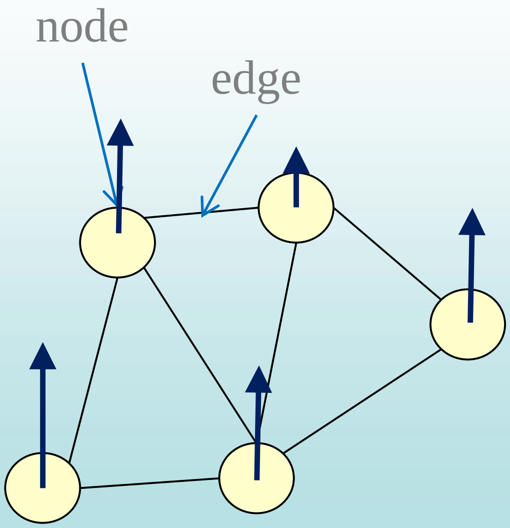
Graph Signal Processing (GSP) addresses the problem of processing signals that live on graphs.

[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*:

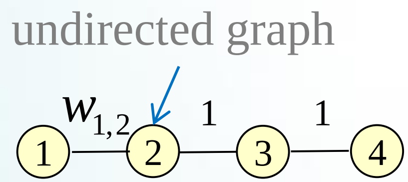
- **Sampling**: how to efficiently acquire / sense a graph-signal?
  - Graph sampling theorems.
- **Representation**: Given graph-signal, how to compactly represent it?
  - Transforms, wavelets, dictionaries.
- **Signal restoration**: Given noisy and/or partial graph-signal, how to recover it?
  - Graph-signal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



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

*Graph Signal Processing Workshop, Pittsburgh, US, May, 2017. https://gsp17.ece.cmu.edu/

Graph Fourier Transform (GFT)



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

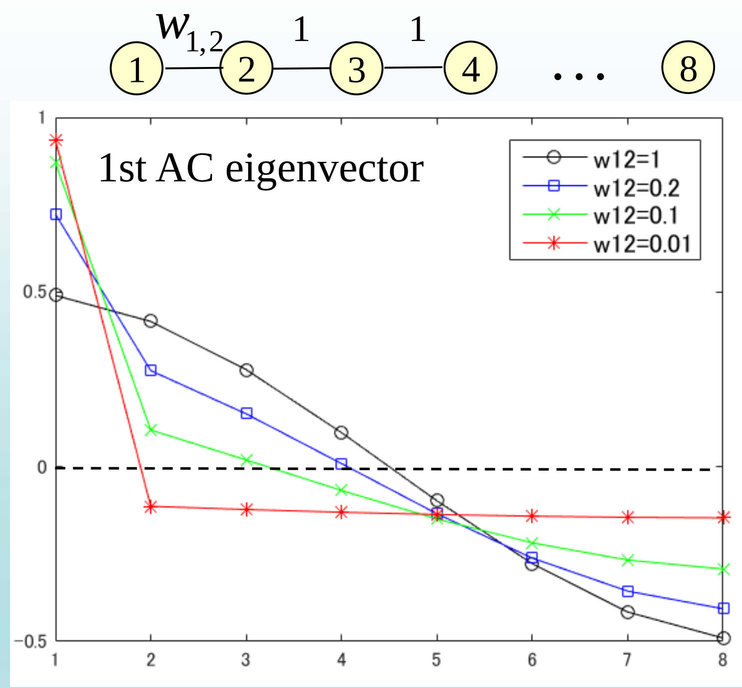
Graph Spectrum from GFT

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

$$L u_i = \lambda_i u_i$$

eigenvalue

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.

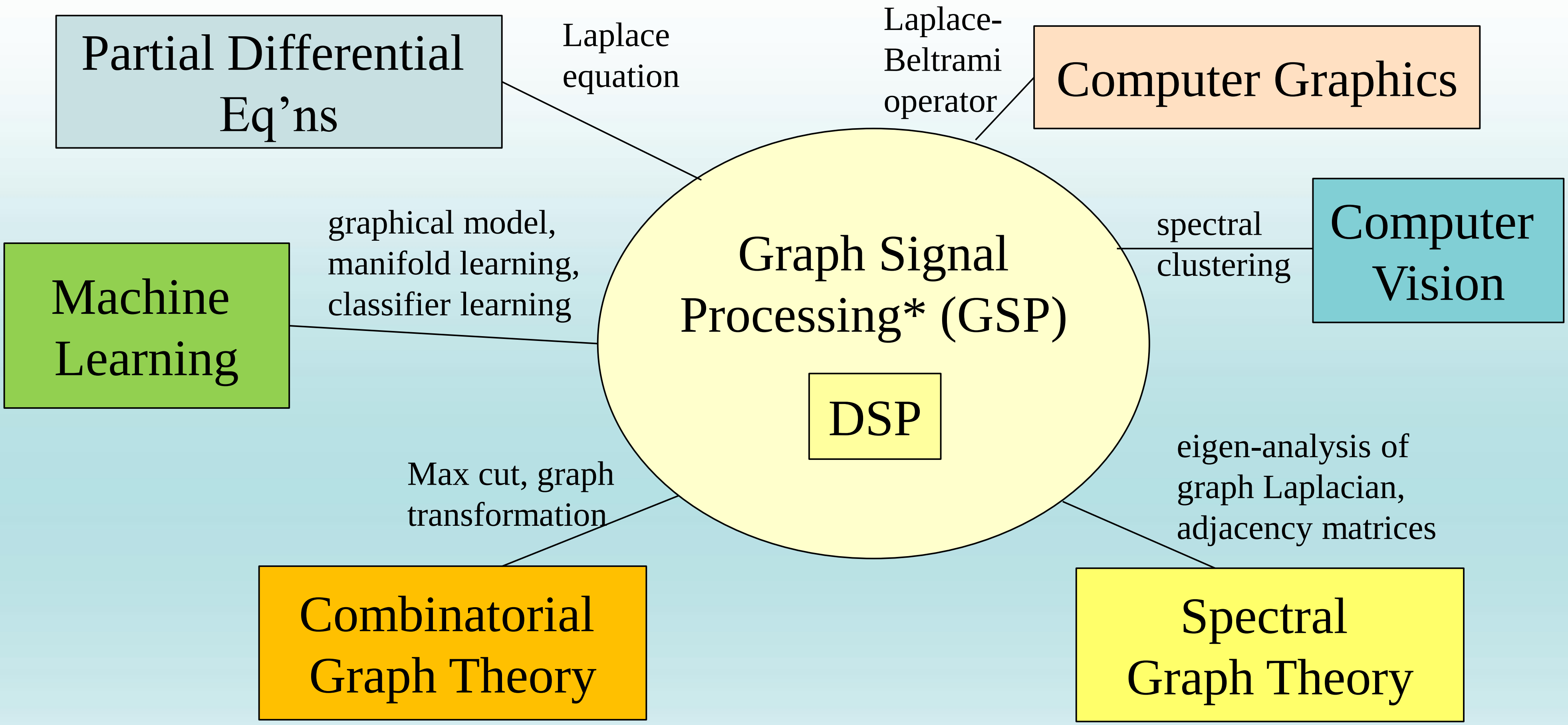
- Normalized.
- Asymmetric.
- Eigenvectors not orthog.

- 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.

GSP and Graph-related Research

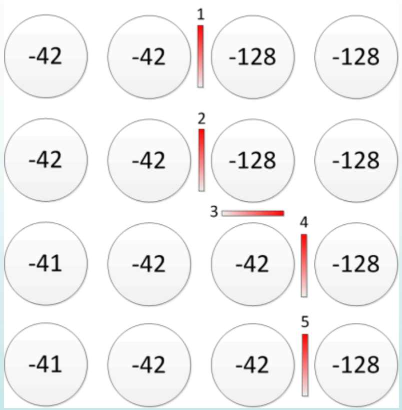
GSP: SP framework that unifies concepts from multiple fields.

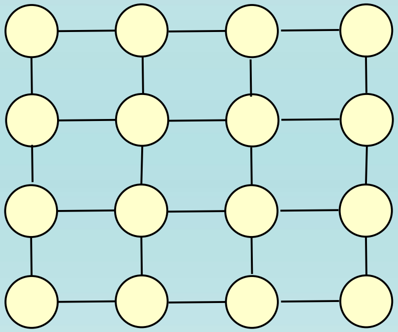


Outline

- Graph Signal Processing
  - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

PWS Image Compression using GFT





- DCT are ***fixed*** basis. Can we do better?
- **Idea**: use ***adaptive*** GFT to improve sparsity [3].

1. Assign edge weight 1 to adjacent pixel pairs.
2. Assign edge weight 0 to sharp signal discontinuity.
3. Compute GFT for transform coding, transmit coeff.

GFT

$$\alpha = \Psi x$$

4. Transmit bits (*contour*) to identify chosen GFT to decoder (overhead of GFT).

Shape-adaptive wavelets can also be done.

[1] G. Shen et al., "**Edge-adaptive Transforms for Efficient Depth Map Coding**," *IEEE Picture Coding Symposium*, Nagoya, Japan, December 2010.

[2] M. Maitre et al., "**Depth and depth-color Coding using Shape-adaptive Wavelets**," *Journal of Visual Communication and Image Representation*, vol.21, July 2010, pp.513-522.

Transform Comparison

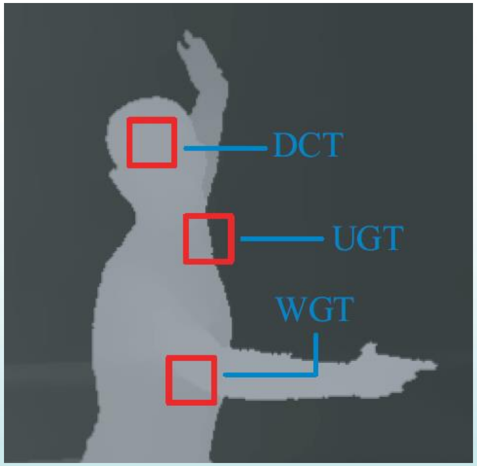
	Transform Representation	Transform Description
Karhunen-Loeve Transform (KLT)	"Sparsest" signal representation given available statistical model	Can be expensive (if poorly structured)
Discrete Cosine Transform (DCT)	*non-sparse signal representation* across sharp boundaries	little (fixed transform)
Graph Fourier Transform (GFT)	minimizes the total rate of signal's transform representation & transform description	

[1] Wei Hu, Gene Cheung, Antonio Ortega, Oscar Au, "**Multiresolution Graph Fourier Transform for Compression of Piecewise Smooth Images**," *IEEE Transactions on Image Processing*, vol.24, no.1, pp.419-433, January 2015.

MR-GFT: Definition of the Search Space for Graph Fourier Transforms

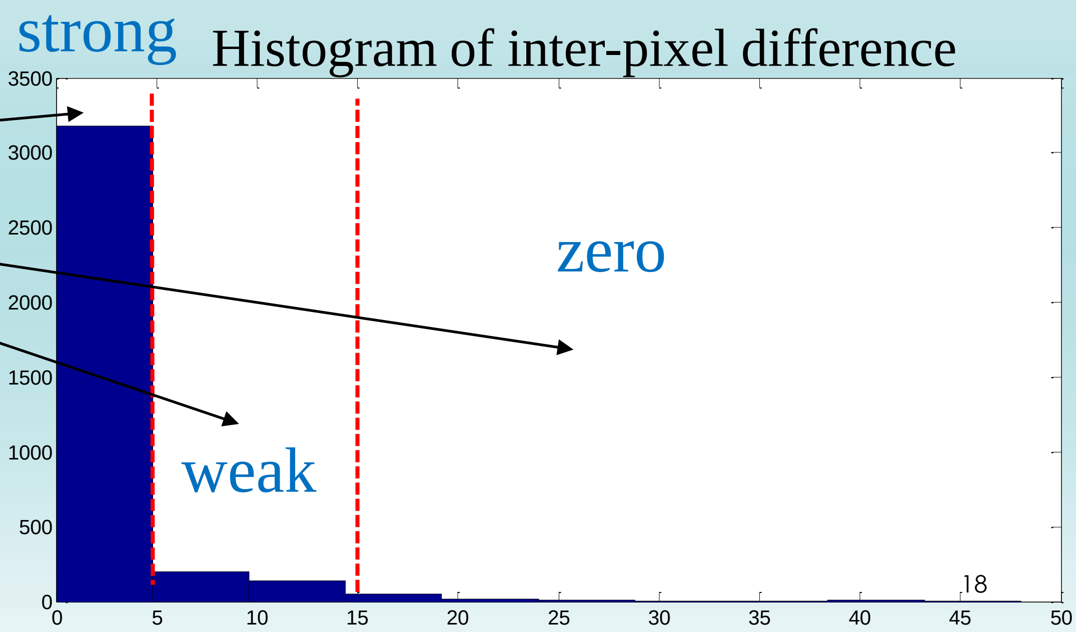
$\boxed{\text{Rate of transform coefficient vector } \alpha}$ $\boxed{\text{Rate of transform description T}}$

$$\min_{\mathbf{W}} \quad R_\alpha(\mathbf{x}, \mathbf{W}) + R_T(\mathbf{W})$$



- In general, weights could be any number in [0,1]
- To limit the description cost R_T
  - Restrict weights to a small discrete set $\mathcal{C} = \{1, 0, c\}$

- "1": *strong correlation* in smooth regions
- "0": *zero correlation* in sharp boundaries
- "c": *weak correlation* in slowly-varying parts



MR-GFT: Derivation of Optimal Edge Weights for Weak Correlation

- Assume a 1D 1st-order ***autoregressive (AR) process*** $\mathbf{x} = [x_1, ..., x_N]^T$ where,

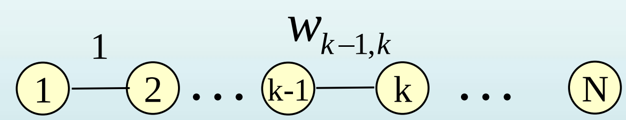
$$x_k = \begin{cases} \eta, & k = 1 \\ x_{k-1} + e_k, & 1 < k \le N, \quad [k-1, k] \in \mathcal{S} \\ x_{k-1} + g + e_k, & 1 < k \le N, \quad [k-1, k] \in \mathcal{P} \end{cases}$$

smooth

jump

non-zero mean random var.

non-zero mean RV



- Assuming the only weak correlation exists between x_{k-1} and x_k

$$\mathbf{F}\mathbf{x} = \mathbf{b},$$

$$\begin{aligned} x_1 &= \eta \\ x_2 - x_1 &= e_2 \\ &\cdots \\ x_k - x_{k-1} &= g + e_k \\ &\cdots \\ x_N - x_{N-1} &= e_N \end{aligned}$$

$$\Rightarrow \quad \mathbf{F} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & \ddots & \ddots & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 0 \\ e_2 \\ \vdots \\ e_k \\ \vdots \\ e_N \end{bmatrix} + \begin{bmatrix} \eta \\ 0 \\ \vdots \\ g \\ \vdots \\ 0 \end{bmatrix}$$

k^{th} row

mean vector

$$\mu = [0 \quad \cdots \quad 0 \quad m_g \quad \cdots \quad m_g]^T$$

k-th

$$\boxed{\mathbf{x} = \mathbf{F}^{-1}\mathbf{b}}$$

MR-GFT: Derivation of Optimal Edge Weights for Weak Correlation (cont'd)

- Covariance matrix

$$
\begin{aligned}
\mathbf{C} &= E[(\mathbf{x}-\mu)(\mathbf{x}-\mu)^T] \\
&= E[\mathbf{x}\mathbf{x}^T] - \mu\mu^T \\
&= E[\mathbf{F}^{-1}\mathbf{b}\mathbf{b}^T(\mathbf{F}^T)^{-1}] - \mu\mu^T \\
&= \mathbf{F}^{-1}E[\mathbf{b}\mathbf{b}^T](\mathbf{F}^T)^{-1} - \mu\mu^T
\end{aligned}
$$

$$
E\left[\mathbf{b}\mathbf{b}^T\right] = \begin{array}{cl}
\begin{bmatrix}
\sigma_1^2 & 0 & & \cdots & 0 & & \cdots & 0 \\
0 & 1 & & \cdots & 0 & & \cdots & 0 \\
 & & \ddots & & & & & \\
0 & 0 & \cdots & 1 & 0 & & \cdots & 0 \\
0 & 0 & \cdots & 0 & \sigma_g^2 + m_g^2 + 1 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 \\
 & & & & & & \ddots & \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1
\end{bmatrix} &
\begin{array}{l}
\\ \\ \\ \\ \text{k-th row} \\ \\ \\ \\
\end{array}
\end{array}
$$

- Precision matrix (tri-diagonal)

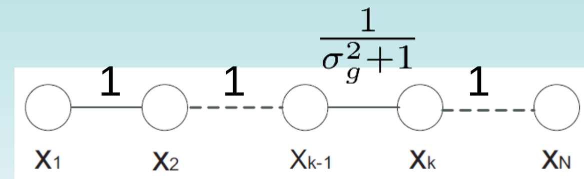
$$
\mathbf{Q} = \mathbf{C}^{-1} = \begin{array}{cl}
\begin{bmatrix}
1+\frac{1}{\sigma_1^2} & -1 & & & & & & \\
-1 & 2 & -1 & & & & & \\
 & \ddots & \ddots & \ddots & & & & \\
 & & -1 & 2 & -1 & & & \\
 & & & -1 & \frac{1}{\sigma_g^2+1}+1 & -\frac{1}{\sigma_g^2+1} & & \\
 & & & & -\frac{1}{\sigma_g^2+1} & \frac{1}{\sigma_g^2+1}+1 & -1 & \\
 & & & & & -1 & 2 & -1 \\
 & & & & & & \ddots & \ddots & \ddots \\
 & & & & & & -1 & 2 & -1 \\
 & & & & & & & -1 & 1
\end{bmatrix} &
\begin{array}{l}
\\ \\ \\ \\ \text{(k-1)-th row} \\ \text{k-th row} \\ \\ \\ \\ \\
\end{array}
\end{array}
\approx \mathcal{L}
$$

$W_{k-1,k}$

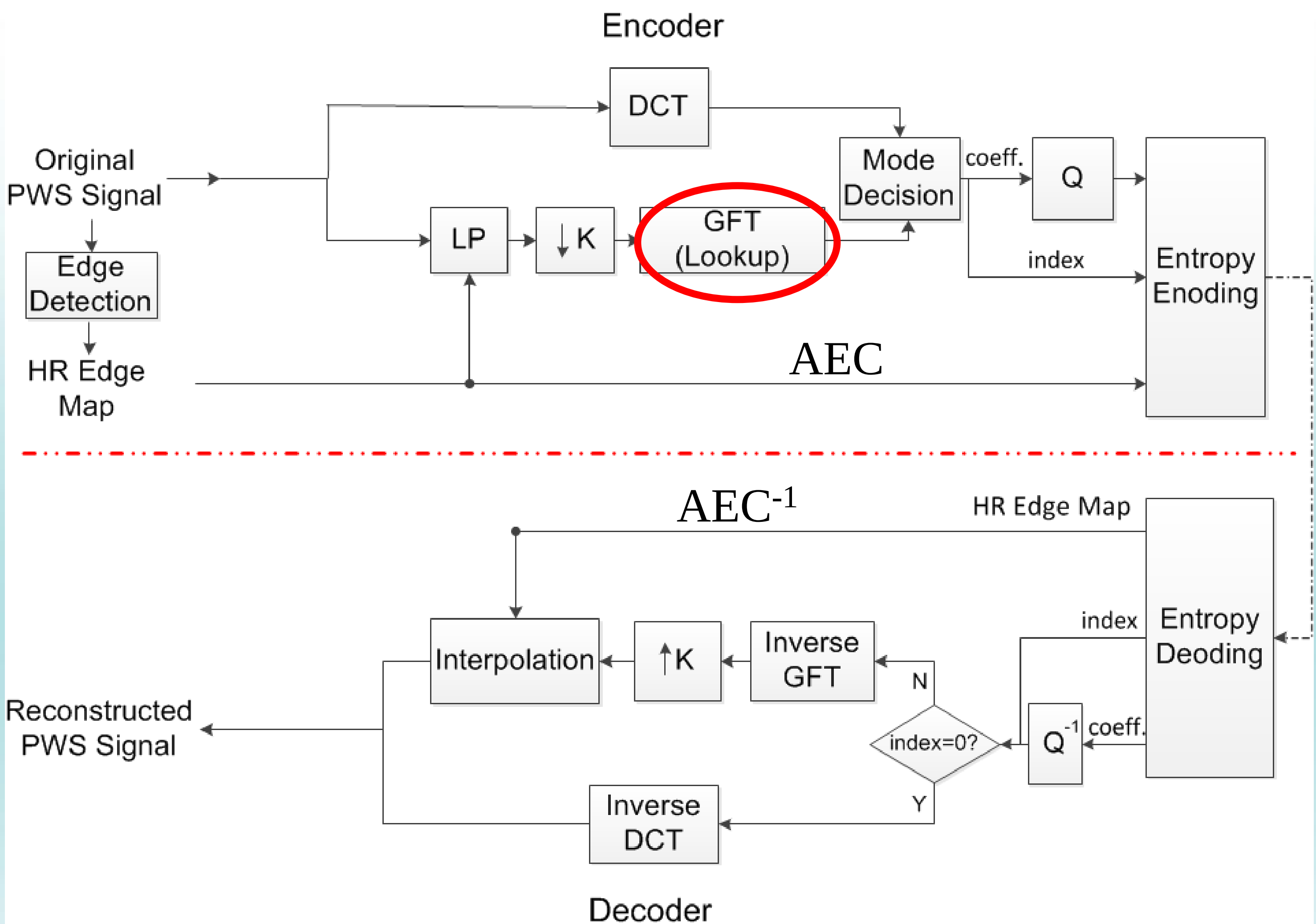
$W_{k,k-1}$

Graph Laplacian matrix!

$$
\boxed{c = W_{k-1,k} = \frac{1}{\sigma_g^2+1}}
$$



MR-GFT: Adaptive Selection of Graph Fourier Transforms



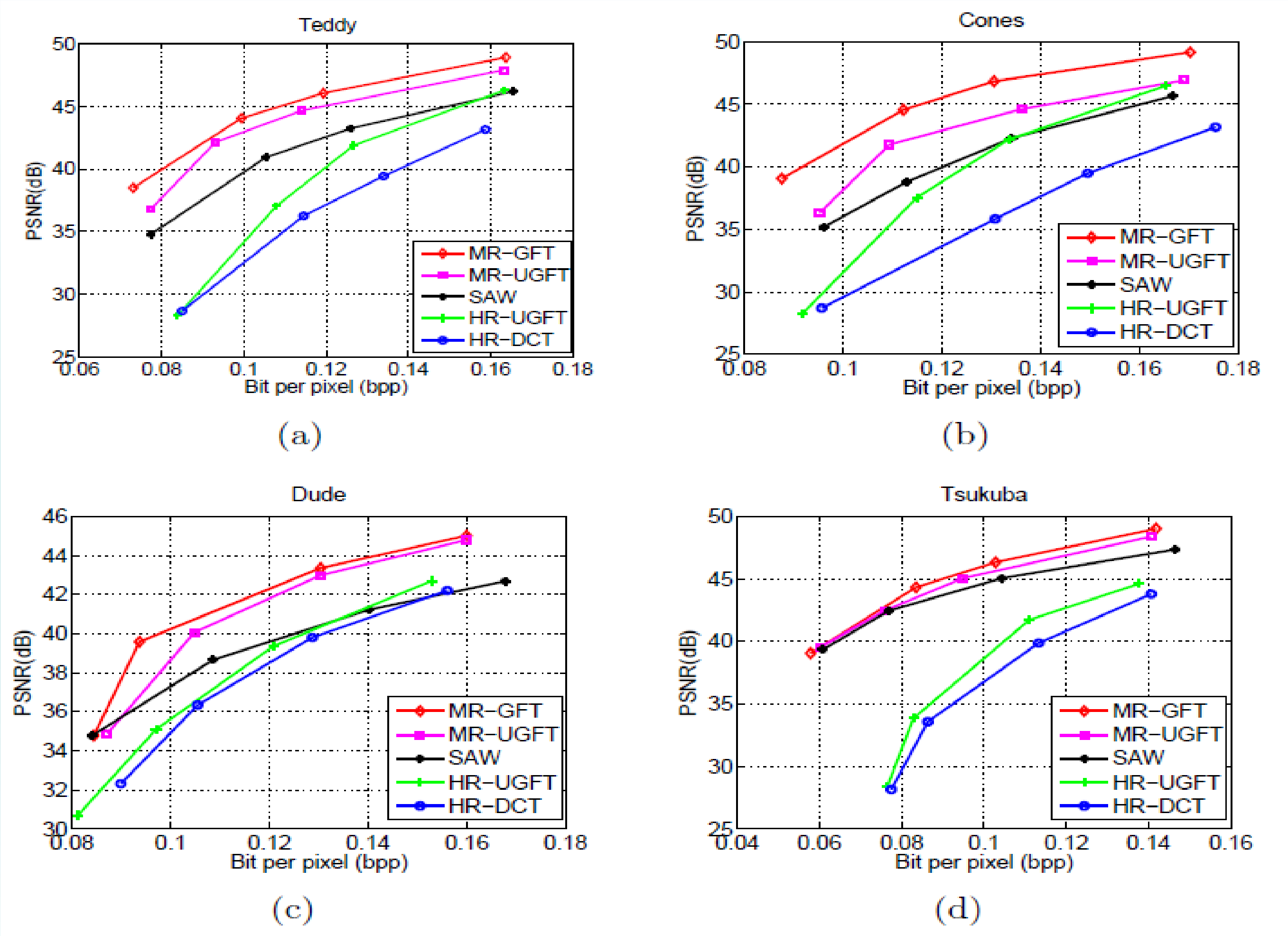
Experimentation

- Setup

  - Test images: depth maps of *Teddy* and *Cones*, and graphics images of *Dude* and *Tsukuba*.

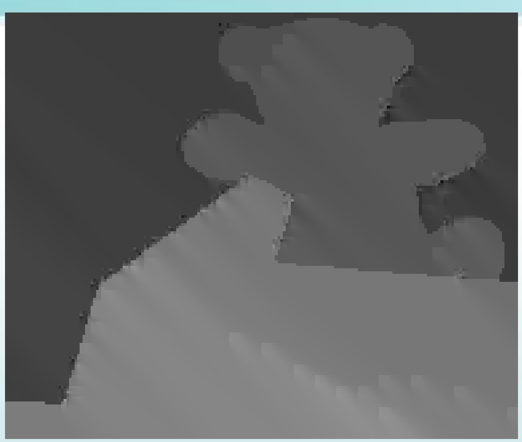
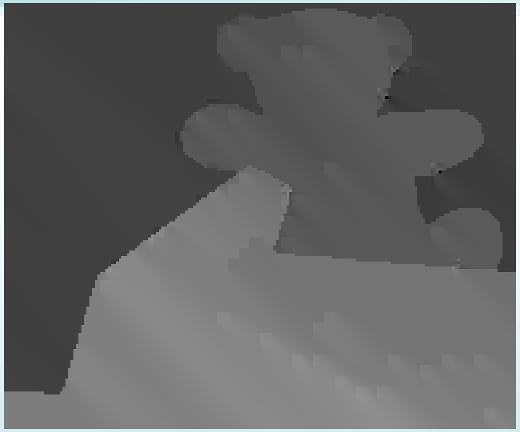
  - Compare against: HR-DCT, HR-SGFT, SAW, MR-SGFT in H.264.

- Results

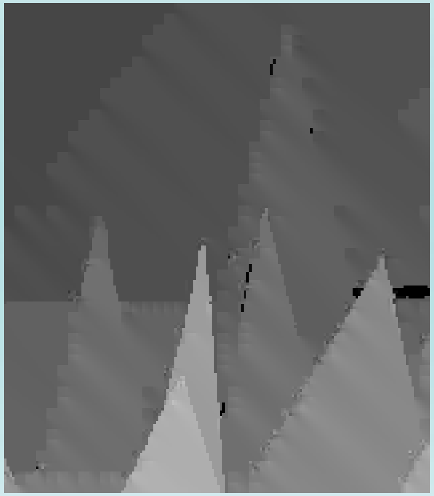
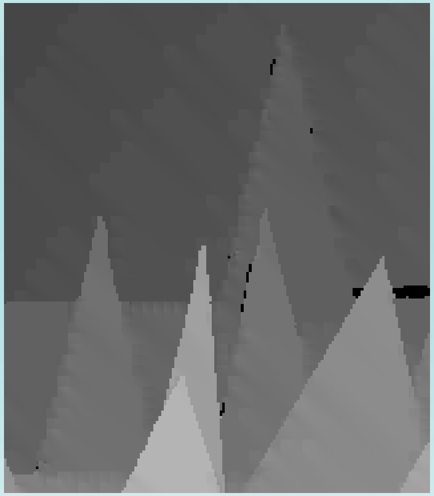
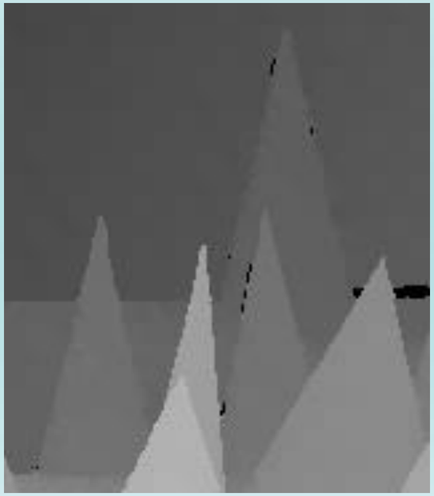


HR-DCT:	6.8dB
HR-SGFT:	5.9dB
SAW:	2.5dB
MR-SGFT:	1.2dB

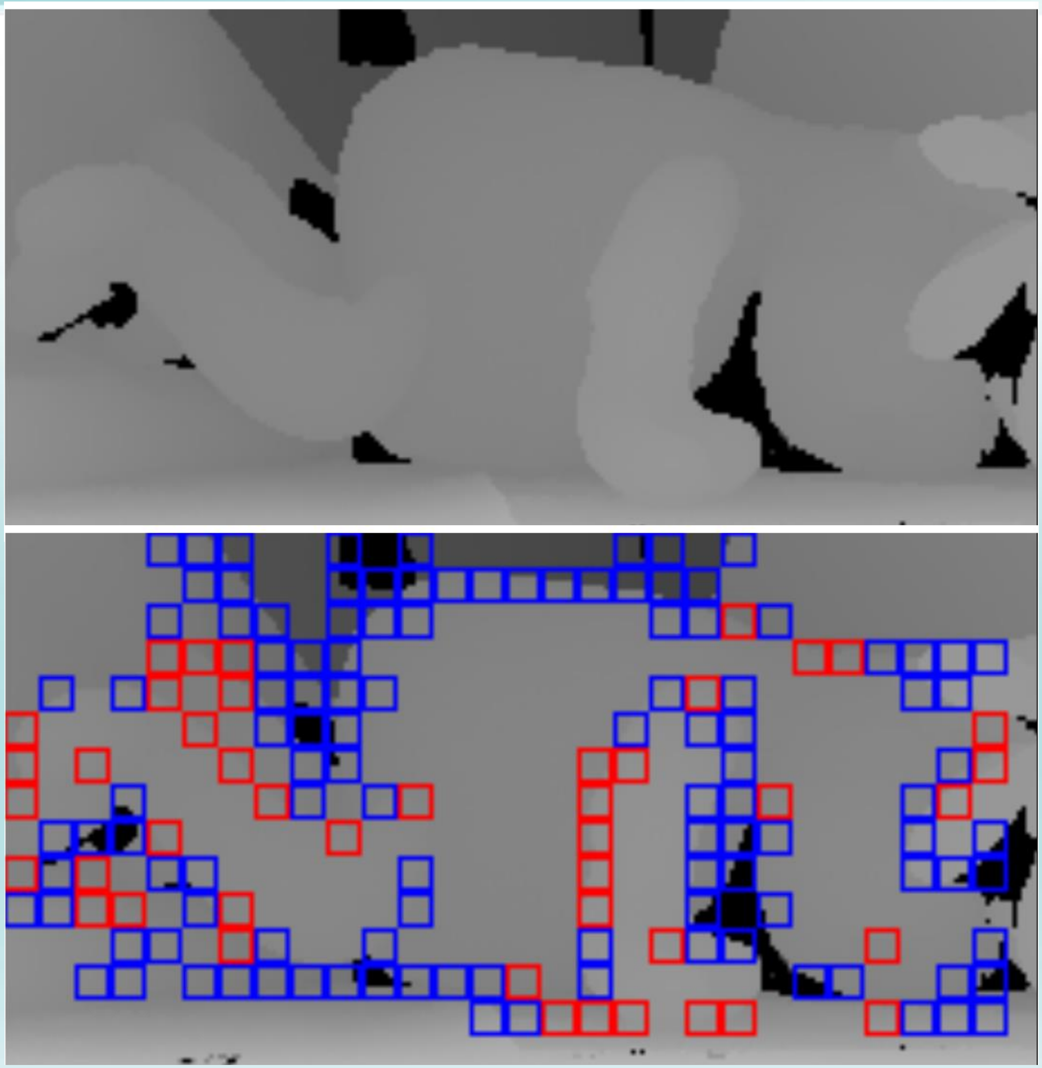
Subjective Results


HR-DCT HR-SGFT MR-GFT

Mode Selection



red: WGFT
blue: UGFT

Edge Coding for PWS Image Compression

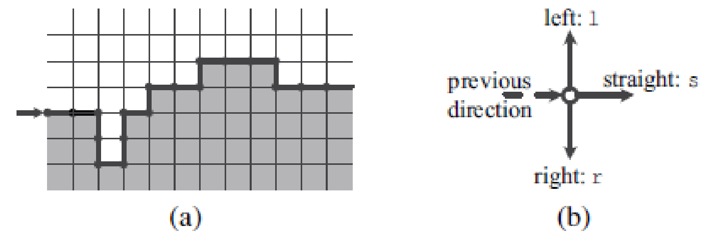
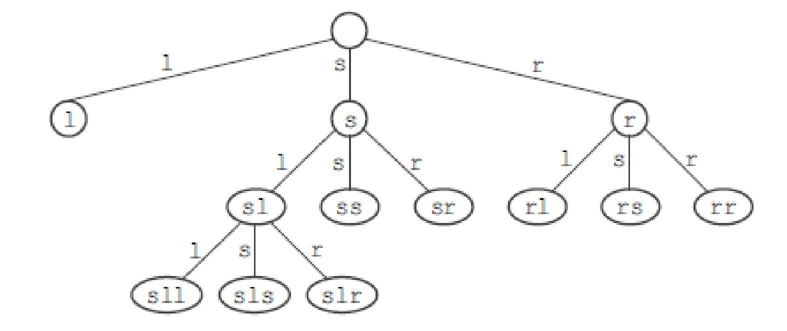
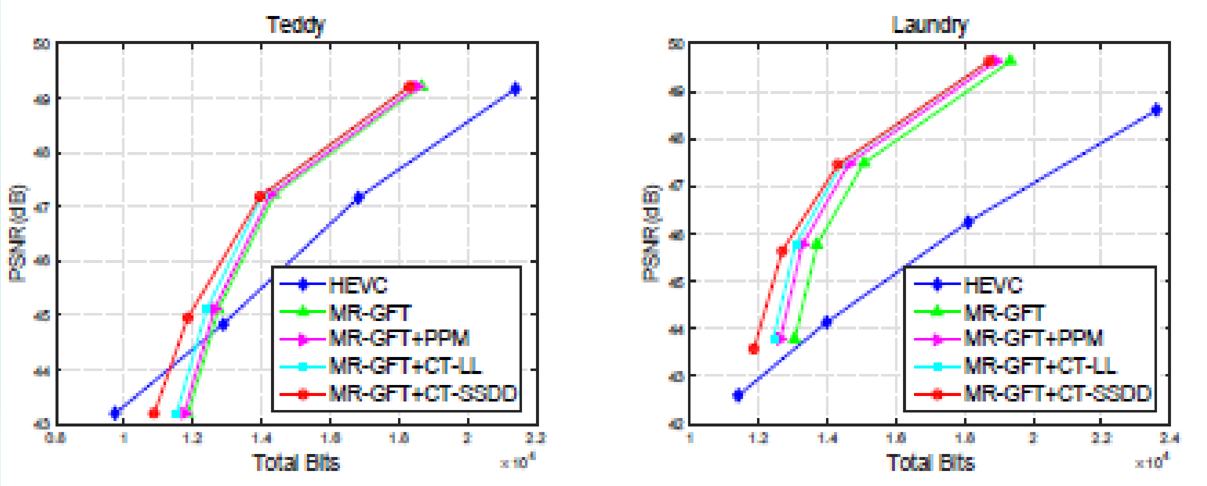


Fig. 2. (a) An example of a contour represented by a four-connected chain codes: east−s−r−s−l−l−s−r−l−r−s−l−r−s−s−r−l−s−s. (b) directional code.

- **Arithmetic Edge Coding** [1,2]:
  - Coding of sequence of *between-pixel edges*, or chain code with symbols {l, s, r}.
  - Design a *variable-length context tree (VCT)* to compute symbol probabilities for arithmetic coding.

$$P(x_i|\mathbf{x}_1^{i-1}) = P(x_i|\mathbf{w})$$





[1] I. Daribo, G. Cheung, D. Florencio, "**Arbitrarily Shaped Sub-block Motion Prediction in Depth Video Compression using Arithmetic Edge Coding**," *IEEE Trans on Image Process*ing, Nov 2014.

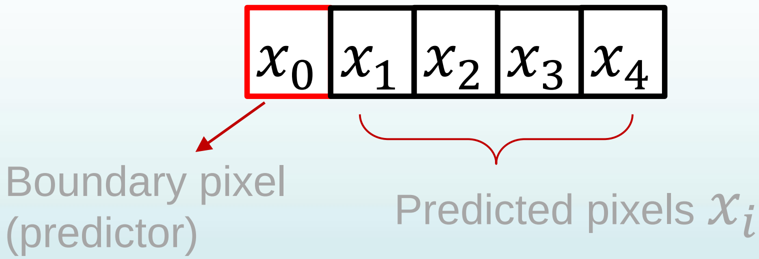
[2] Amin Zheng, Gene Cheung, Dinei Florencio, "**Context Tree based Image Contour Coding using A Geometric Prior**," *IEEE Transactions on Image Processing*, vol.26, no.2, pp.574-589, February 2017.

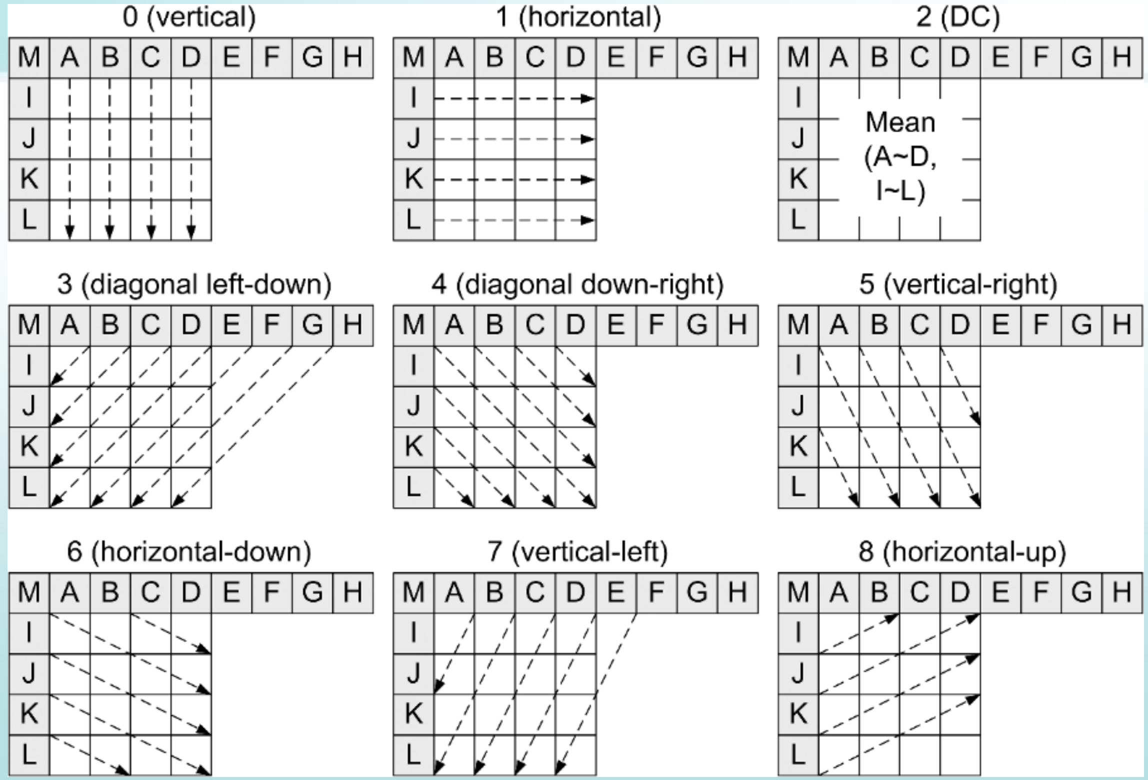
Outline

- Graph Signal Processing
  - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

Motivation

- **Intra-prediction**





$x_i - x_0$: prediction residuals

Intra-prediction in H.264

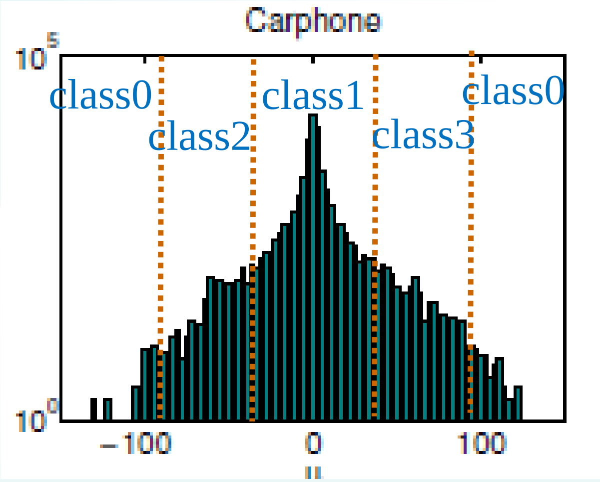
- Discontinuities at block boundaries

  - intra-prediction will not be chosen or bad prediction

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

Optimal 1D Intra prediction

Histogram of inter-pixel difference



Assume a 1D 1st-order ***autoregressive (AR) process***

$$x_n = x_{n-1} + \hat{\mu}_{i(\mu_n)} + g_{i(\mu_n)}$$

bin average

approximation error

$$\begin{bmatrix} x_0 + \hat{\mu}_a \\ \vdots \\ x_0 + \hat{\mu}_a \\ x_0 + \hat{\mu}_a + \hat{\mu}_b \\ \vdots \\ x_0 + \hat{\mu}_a + \hat{\mu}_b \end{bmatrix}$$

Class k

Class l

- Optimal prediction in terms of resulting in a *zero-mean* prediction residual
- Default to conventional intra-prediction when $\hat{\mu}_a = \hat{\mu}_b = 0$, i.e.,

$$[x_0, \ldots, x_0]^T$$

Generalized Graph Fourier Transform

- The precision matrix of the prediction residual

$$\begin{bmatrix} \alpha_a+1 & -1 & & & & & & \\ -1 & 2 & -1 & & & & & \\ & \ddots & \ddots & \ddots & & & & \\ & & -1 & 1+\alpha_b & -\alpha_b & & & \\ & & & -\alpha_b & \alpha_b+1 & -1 & & \\ & & & & \ddots & \ddots & \ddots & \\ & & & & & -1 & 2 & -1 \\ & & & & & & -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -1 & & & & & & \\ -1 & 2 & -1 & & & & & \\ & \ddots & \ddots & \ddots & & & & \\ & & -1 & 1+\alpha_b & -\alpha_b & & & \\ & & & -\alpha_b & \alpha_b+1 & -1 & & \\ & & & & \ddots & \ddots & \ddots & \\ & & & & & -1 & 2 & -1 \\ & & & & & & -1 & 1 \end{bmatrix} + \begin{bmatrix} \alpha_a & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \end{bmatrix}$$

Generalized Laplacian — Combinatorial Laplacian — Degree matrix for boundary vertices

- Default to the DCT if $\alpha_a = 0$ and $\alpha_b = 1$
- Default to the ADST [1] if $\alpha_a = 1$ and $\alpha_b = 1$

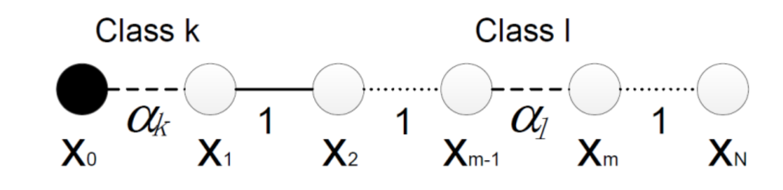
inaccuracy of intra-prediction

$$\alpha_a = \sigma_{g_0}^2 / \sigma_{g_a}^2$$

discontinuities within signal

$$\alpha_b = \sigma_{g_0}^2 / \sigma_{g_b}^2$$

Variance of approx. error

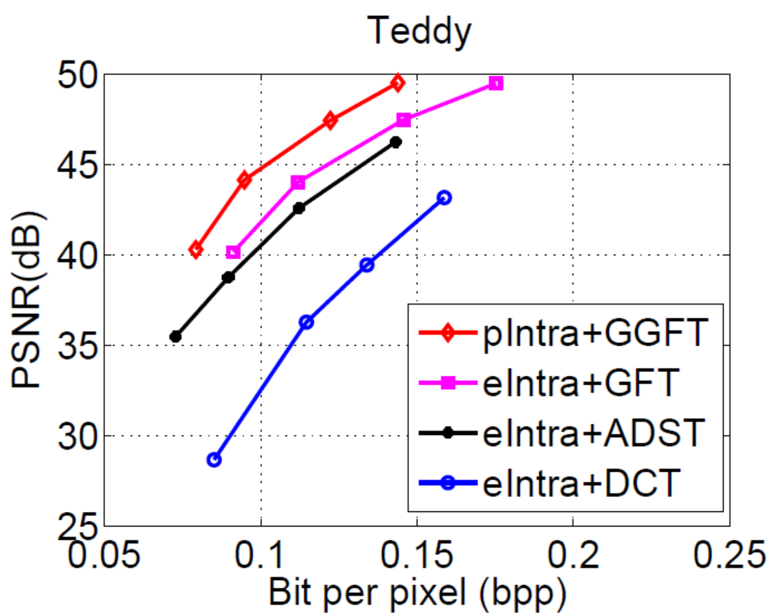


[1] J. Han et al., "**Jointly Optimized Spatial Prediction and Block Transform for video and Image Coding**," *IEEE Transactions on Image Processing*, vol.21, no.4, April 2012, pp.1874-1884.

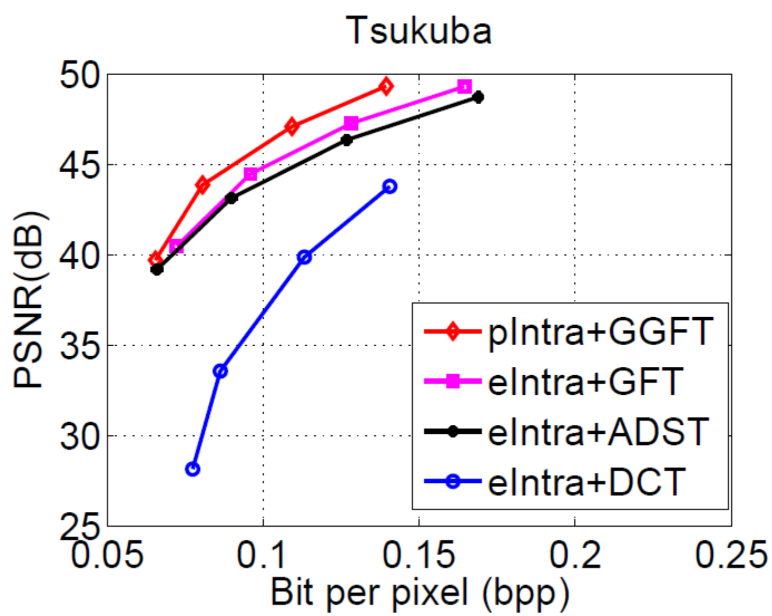
Experimental Results

- Test images: PWS images and natural images
- Compare *proposed intra-prediction (pIntra)* + *GGFT* against:

  - edge-aware intra-prediction (eIntra) + DCT
  - eIntra + ADST
  - eIntra + GFT



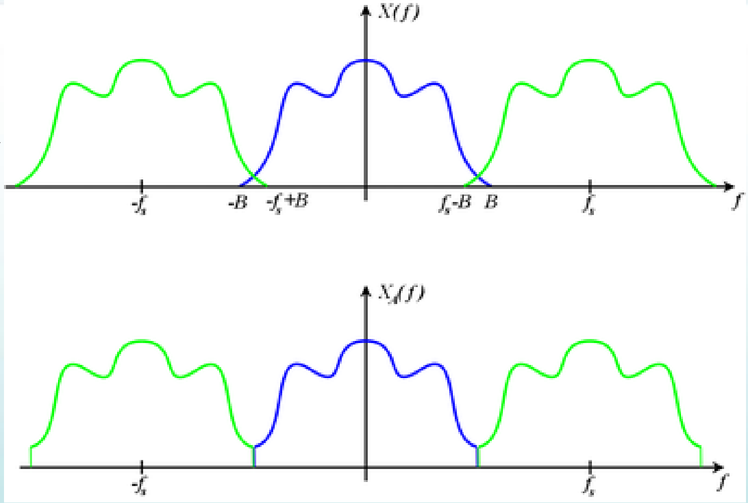
(a) Teddy



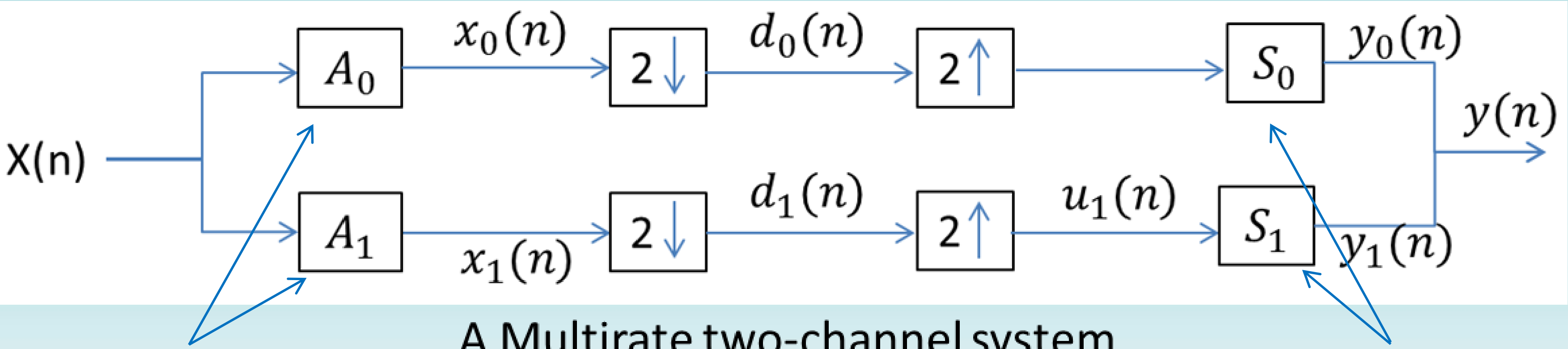
(b) Tsukuba

Spectral Folding & Critical Sampling

- **Spectral Folding**:
  - (Sub)sampling a bandlimited signal at freq. f_s → freq. content replication at f_s.
- **Nyquist Sampling Theorem**:
  - To avoid aliasing, sample at 2x max. freq. of bandlimited signal.



- Multirate Wavelet Filterbank:
  - System of "perfect reconstruction" bandpass filters



A Multirate two-channel system

analysis filter

synthesis filter

Bipartite Graph Approximation

- **Problem**: GraphBior [1,2] (critically sampled, perfect reconst. wavelet) for ***bipartite graph*** only!
- **Idea** [3]:
  - Successively find **bipartite graph approximation**.
  - Criteria for graph approx [1]:

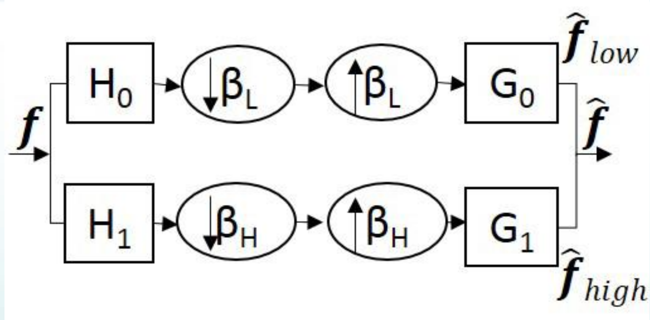
$$\min_{L^b} D_{KL}\left(L \,\|\, L^b\right) - \gamma \,\mathrm{rank}\left(L^b_{1,2}\right)$$

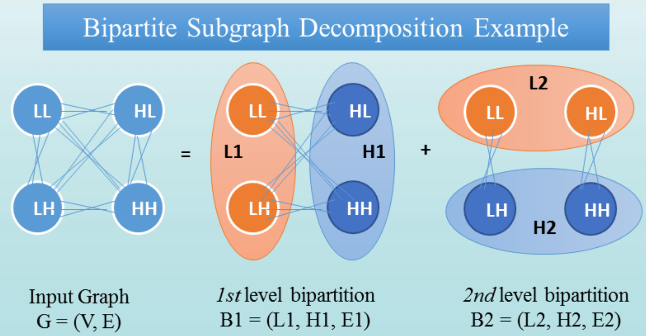
bipartite graph Laplacian

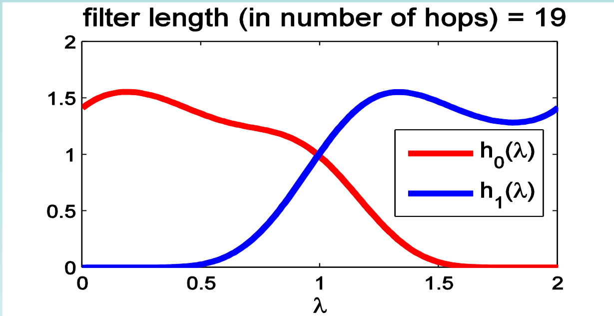
KL divergence

sub-matrix for 2 partitions

    - Preserve graph structure, minimize eigenvalue=1.







[1] S. Narang and A. Ortega, "**Perfect reconstruction two-channel wavelet filter banks for graph structured data**," *IEEE Transactions on Signal Processing*, vol. 60, no. 6,pp. 2786–2799, June 2012.

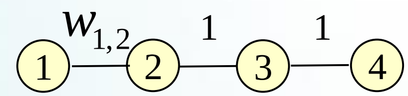
[2] S. Narang and A. Ortega, "**Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs**," *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4673–4685, Oct 2013.

[3] Jin Zeng, Gene Cheung, Antonio Ortega, "**Bipartite Subgraph Decomposition for Critically Sampled Wavelet Filterbanks on Arbitrary Graphs**," *IEEE International Conference on Acoustics, Speech and Signal Processing*, Shanghai, China, March, 2016.

Outline

- Graph Signal Processing
  - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

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 $\longrightarrow$ $y = x + v$ $\longleftarrow$ noise

- MAP formulation:

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

fidelity term

smoothness prior

[1] P. Milanfar, "**A Tour of Modern Image Filtering: New Insights and Methods, Both Practical and Theoretical**," *IEEE Signal Processing Magazine*, vol.30, no.1, pp.106-128, January 2013.

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.

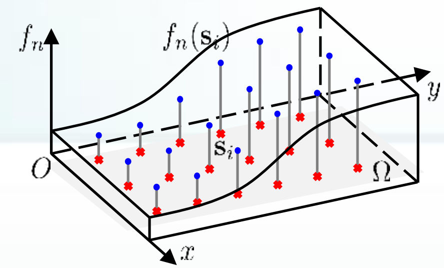
TABLE I

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	-

[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.

Analysis of Graph Laplacian Regularizer



- Show $S_G(\mathbf{u}) = \mathbf{u}^{\mathrm{T}}\mathbf{L}\mathbf{u}$ converges to continuous functional S_Ω, analysis of S_Ω explains how $\mathbf{u}^{\mathrm{T}}\mathbf{L}\mathbf{u}$ penalizes candidates:

$$\mathrm{prior}(\mathbf{x}) = \mathbf{x}^T \mathbf{L}\,\mathbf{x} \quad \rightarrow \; S_\Omega(x) = \int_\Omega \nabla x^T \mathbf{G}^{-1} \nabla x \left(\sqrt{\det \mathbf{G}}\right)^{2\gamma-1} ds$$

feature function vector

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

- Derive optimal $S_G(\mathbf{u}) = \mathbf{u}^{\mathrm{T}}\mathbf{L}\mathbf{u}$ for denoising: graph is discriminant for small noise, robust when very noisy.

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

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

Non-local self-similarity and MMSE formulation	$\xrightarrow{\text{obtain}}$	Optimal regularizer S_G

metric space $\mathbf{G} = \sum_{n=1}^{N} \nabla f_n \cdot \nabla f_n^{\mathrm{T}}$

- We interpret graph Laplacian regularization as anisotropic diffusion, show that it not only smooths but may also sharpens the image, promote piecewise smooth images

[1] J. Pang, G. Cheung, "**Graph Laplacian Regularization for Inverse Imaging: Analysis in the Continuous Domain**," *IEEE Transactions on Image Processing*, vol. 26, no.4, pp.1770-1785, April 2017.

Denoising Experiments (natural images)

- Subjective comparisons ($\sigma_I = 40$)

Original

Noisy, 16.48 dB

K-SVD, 26.84 dB

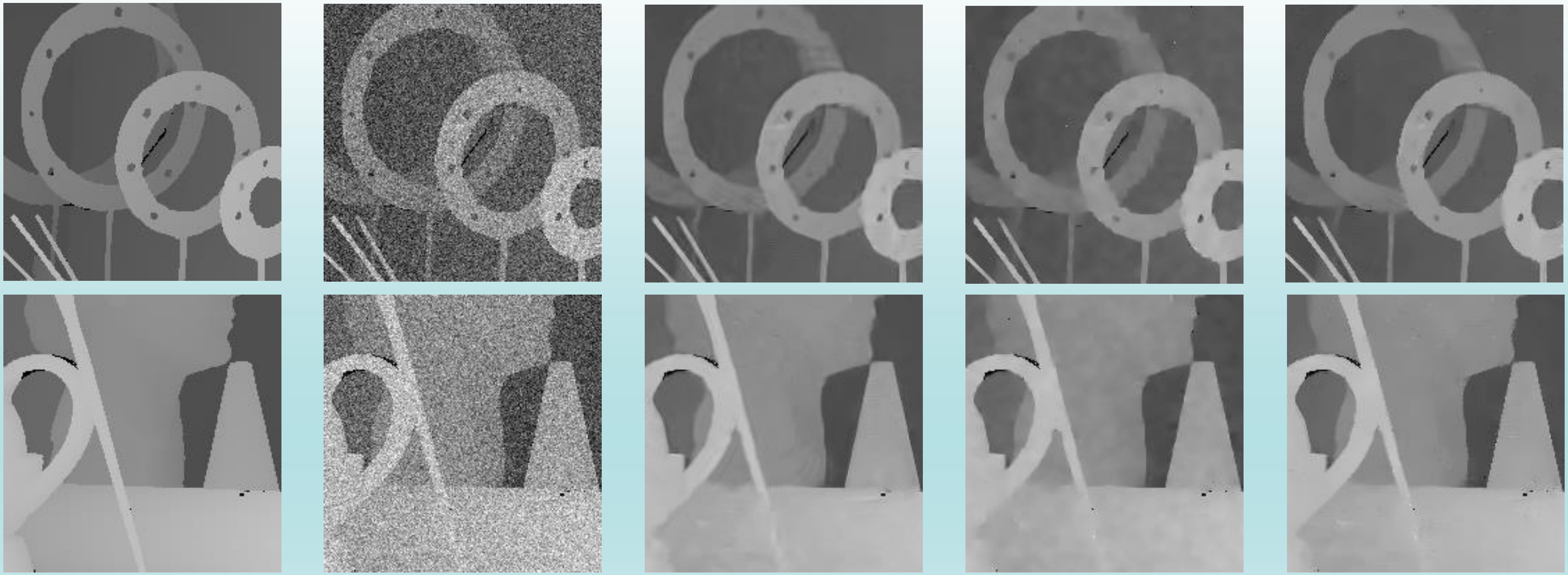
BM3D, 27.99 dB

PLOW, 28.11 dB

OGLR, 28.35 dB

Denoising Experiments (depth images)

- Subjective comparisons ($\sigma_I = 30$)



Original | Noisy, 18.66 dB | BM3D, 33.26 dB | NLGBT, 33.41dB | OGLR, 34.32 dB

[1] W. Hu et al., "**Depth Map Denoising using Graph-based Transform and Group Sparsity**," *IEEE International Workshop on Multimedia Signal Processing*, Pula (Sardinia), Italy, October, 2013.

Outline

- Graph Signal Processing
  - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

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:

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

DCT

8x8 pixel block

quantization parameter

DCT Coefficients

  3. Quantized DCT coeff entropy coded.
- **Decoder**: uncertainty in signal reconstruction:

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

[1] A. Zakhor, "**Iterative procedures for reduction of blocking effects in transform image coding**," *IEEE Transactions on Circuits and Systems for Video Technology*,, vol. 2, no. 1, pp. 91–95, Mar 1992.

[2] K. Bredies and M. Holler, "**A total variation-based JPEG decompression model**," *SIAM J. Img. Sci.*, vol. 5, no. 1, pp. 366–393, Mar. 2012.

[3] H. Chang, M. Ng, and T. Zeng, "**Reducing artifacts in jpeg decompression via a learned dictionary**," *IEEE Transactions on Signal Processing*,,vol. 62, no. 3, pp. 718–728, Feb 2014.

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?

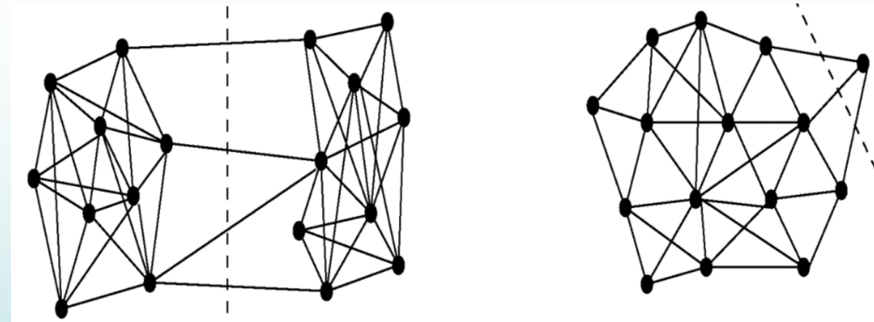
[1] W. Hu, G. Cheung, M. Kazui, "**Graph-based Dequantization of Block-Compressed Piecewise Smooth Images**," *IEEE Signal Processing Letters*, vol.23, no.2, pp.242-246, February 2016.

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

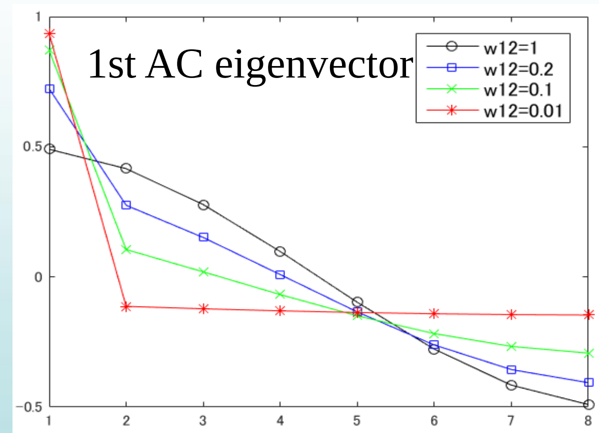
[1] J. Shi and J. Malik, “**Normalized cuts and image segmentation**,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

Eigenvectors of Normalized graph Laplacian

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

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

candidate objective



- Define:

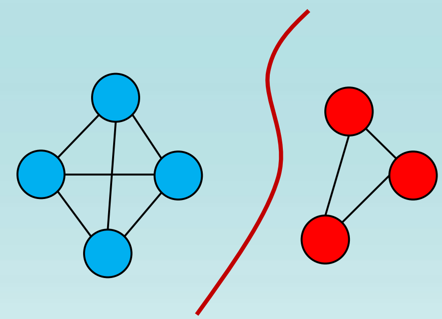
$$\mathrm{v} := \mathrm{D}^{1/2}\mathrm{f} \qquad \mathrm{v}_1 := \mathrm{D}^{1/2}\mathrm{1}$$

- Problem rewritten as:

Rayleigh quotient

$$\mathrm{v}^* = \arg\min_{\mathrm{v}} \frac{\mathrm{v}^T \mathrm{L}_\mathrm{n} \mathrm{v}}{\mathrm{v}^T \mathrm{v}} \quad s.t.\ \mathrm{v}^T \mathrm{v}_1 = 0$$

  - $\mathbf{v}_1$ 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.

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]:

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

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

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

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

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

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

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

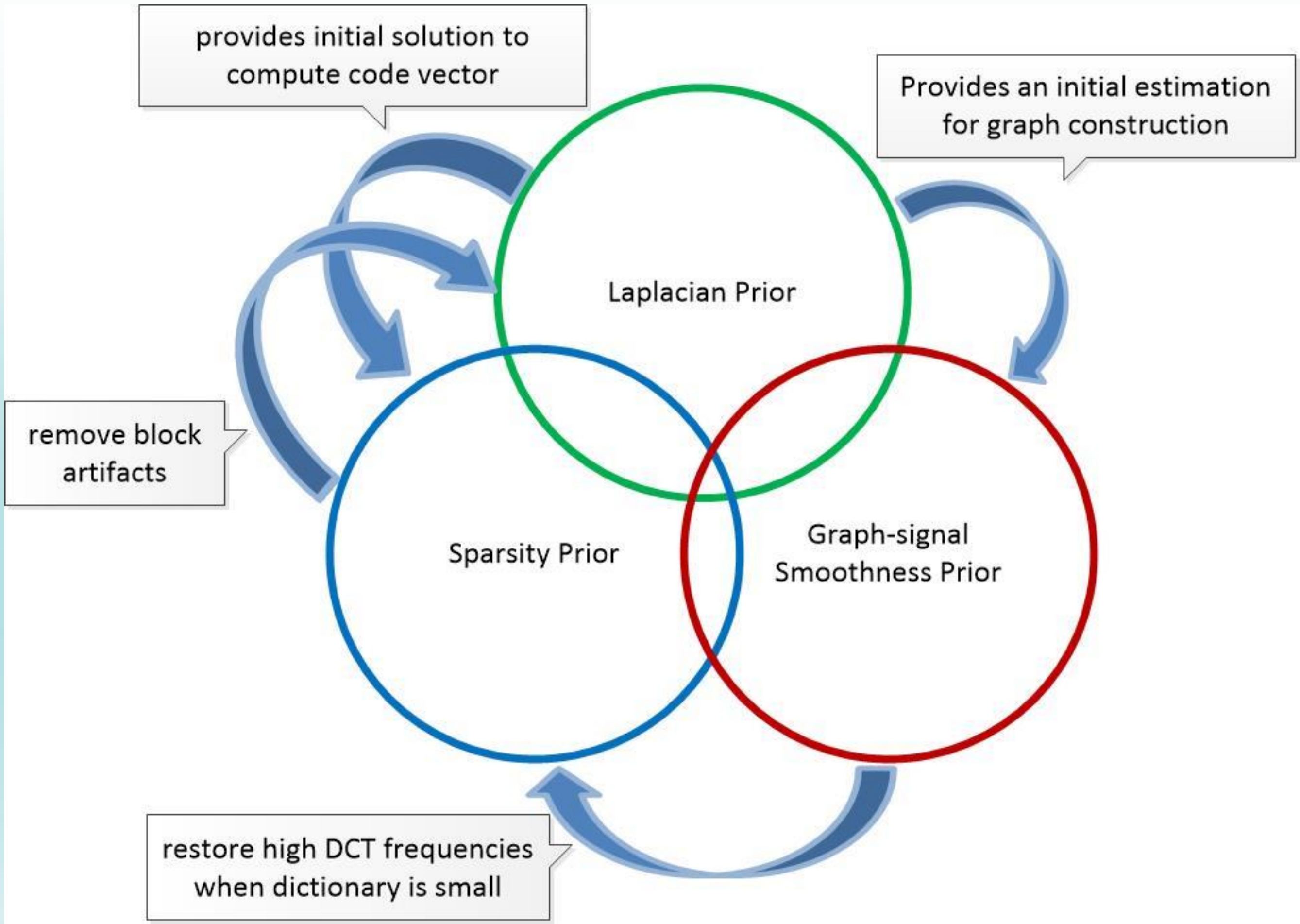
[1] A. Kheradmand and P. Milanfar, "**A general framework for regularized, similarity-based image restoration**," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5136–5151, Dec 2014.

LERaG for 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



[1] Xianming Liu, Gene Cheung, Xiaolin Wu, Debin Zhao, "**Random Walk Graph Laplacian based Smoothness Prior for Soft Decoding of JPEG Images**," *IEEE Transactions on Image Processing*, vol.26, no.2, pp.509-524, February 2017.

Soft Decoding Algorithm w/ Prior Mixture

- Objective:

fidelity term

sparsity prior

graph-signal smoothness prior

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

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

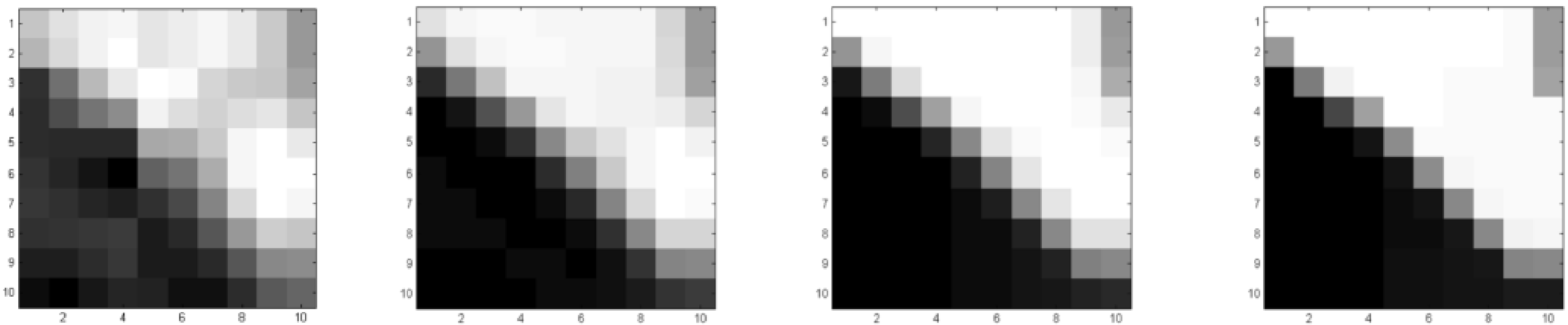
graph-signal, code vector

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**.

Evolution of 2nd Eigenvector

- 2nd Eigenvector becomes more PWS:



(a) initialization, LERaG = 76453.02, 2nd Eigenvalue = 0.001079
(b) iter = 1, LERaG = 64827.99, 2nd Eigenvalue = 315e-6
(c) iter = 2, LERaG = 14057.09, 2nd Eigenvalue = 35e-6
(d) iter = 3, LERaG = 6440.29, 2nd Eigenvalue = 3e-6

- PWS means:
  1. better pixel clusters,
  2. smaller Fidler number (2nd eigenvalue),
  3. Smaller smoothness penalty term.

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**

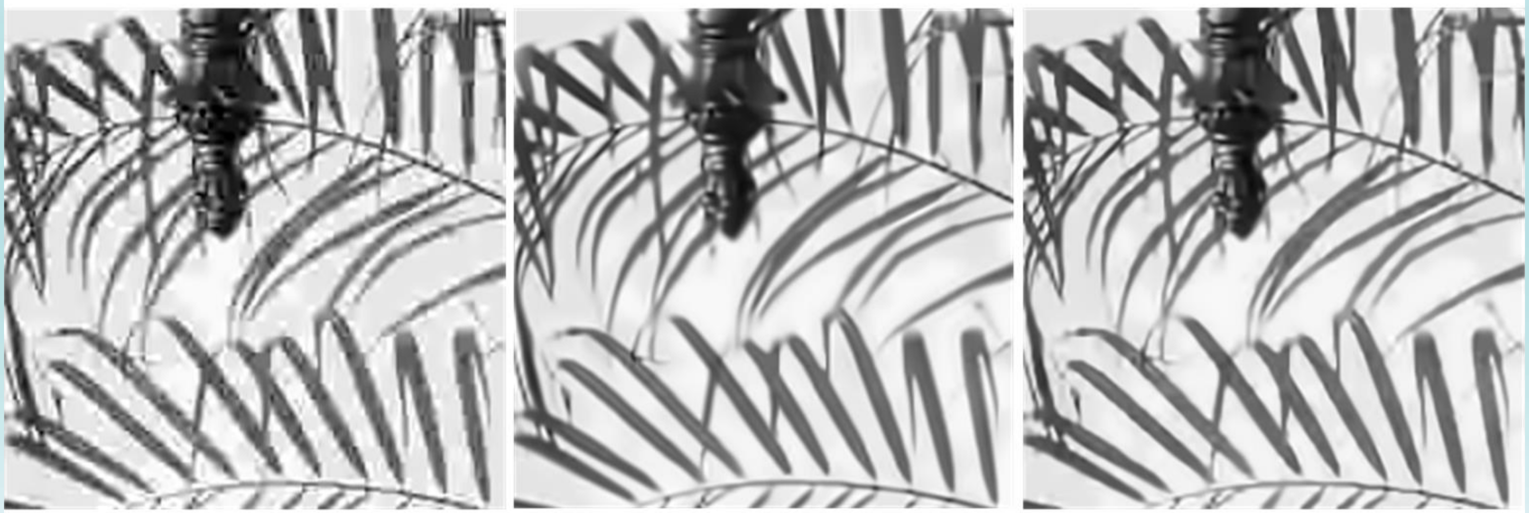
Subjective Quality Evaluation

(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 (25.31,0.8704) (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

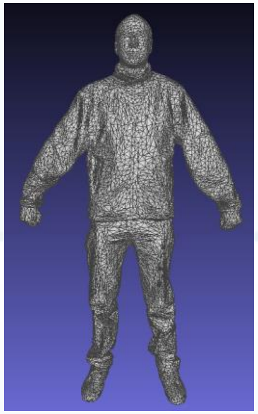
- Comparisons w/ other graph regularizers:

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**

Outline

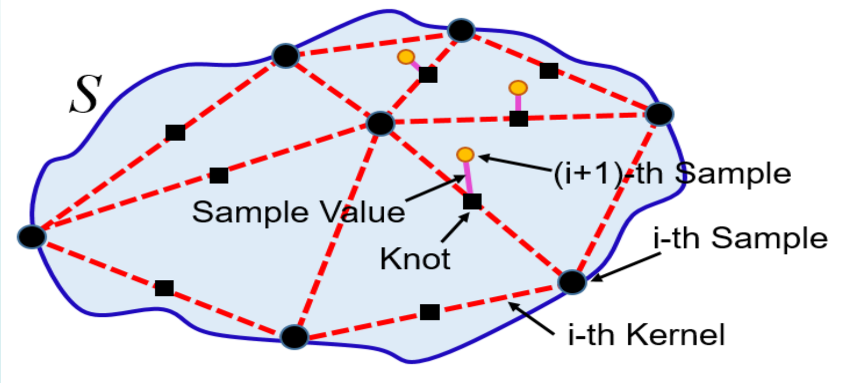
- Graph Signal Processing
    - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

Graph-Signal Sampling / Encoding for 3D Point Cloud



MIT *dataset**

- **Problem**: Point clouds require encoding specific 3D coordinates.
- **Assumption**: smooth 2D manifold in 3D space.
- **Proposal**: progressive 3D geometry rep. as series of graph-signals.
  1. adaptively identifies new samples on the manifold surface, and
  2. encodes them efficiently as graph-signals.

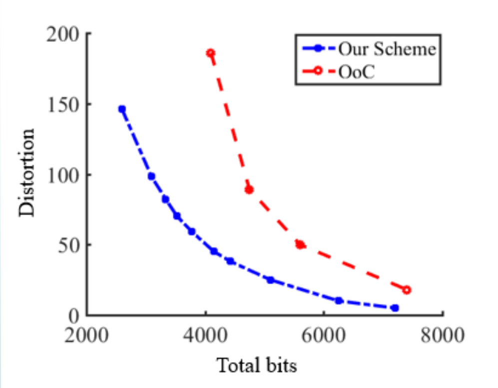


- **Example**:

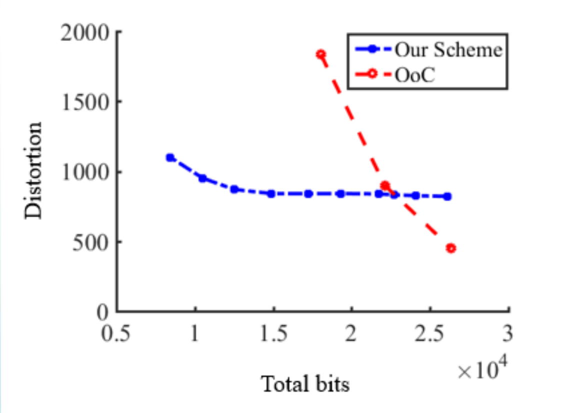
1. Interpolate i^{th} iteration samples (black circles) to a **continuous kernel** (mesh), an approximation of the target surface ***S***.
2. New sample locations, **knots** (squares), are located on the kernel surface.
3. **Signed distances** between knots and ***S*** are recorded as sample values.
4. **Sample values** (green circles) are encoded as a **graph-signal via GFT**.

Graph-Signal Sampling / Encoding for 3D Point Cloud

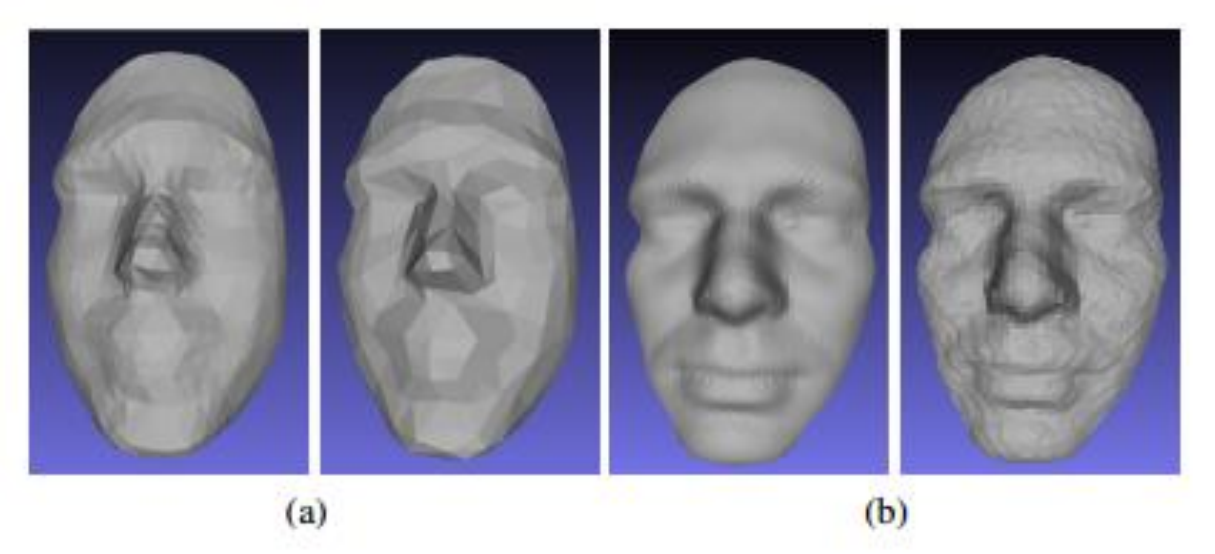
- **Experimental Results**:



(a) Dataset1



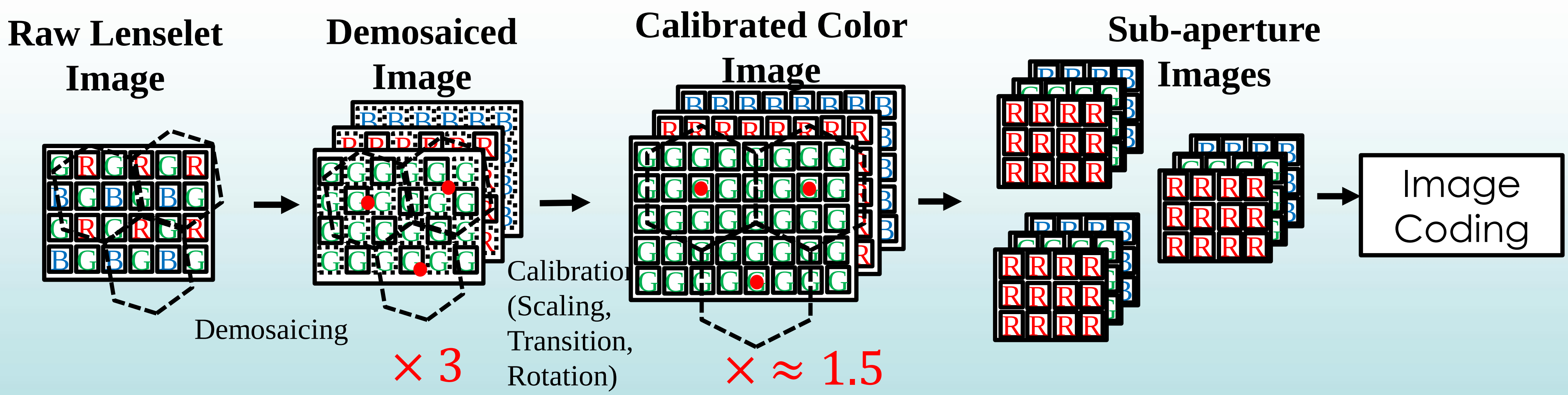
(b) Dataset2



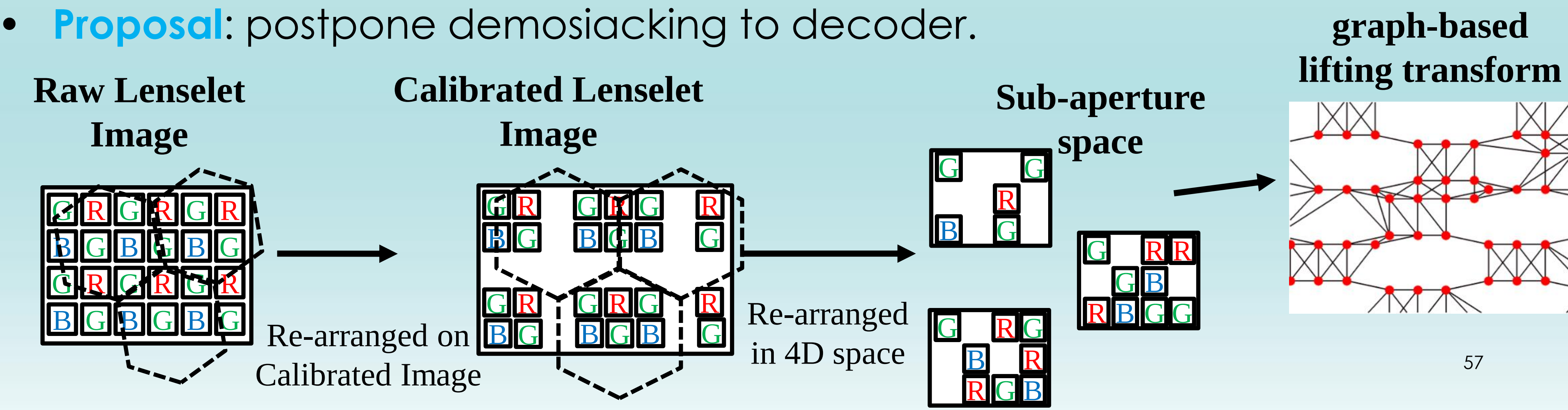
[1] Mingyuan Zhao, Gene Cheung, Dinei Florencio, Xiangyang Ji, "**Progressive Graph-Signal Sampling and Encoding for Static 3D Geometry Representation**," *IEEE International Conference on Image Processing*, Beijing, China, September, 2017.

Pre-Demosiac Light Field Image Compression Using Graph Lifting Transform

- **Problem**: Sub-aperture images in Light field data are huge.



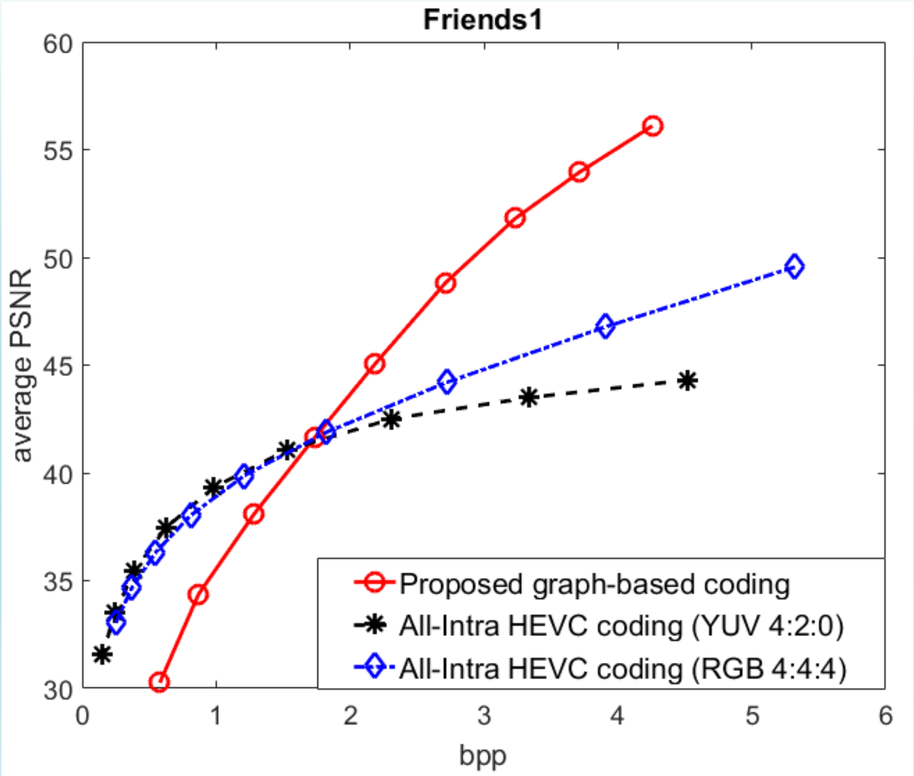
- **Proposal**: postpone demosiacking to decoder.

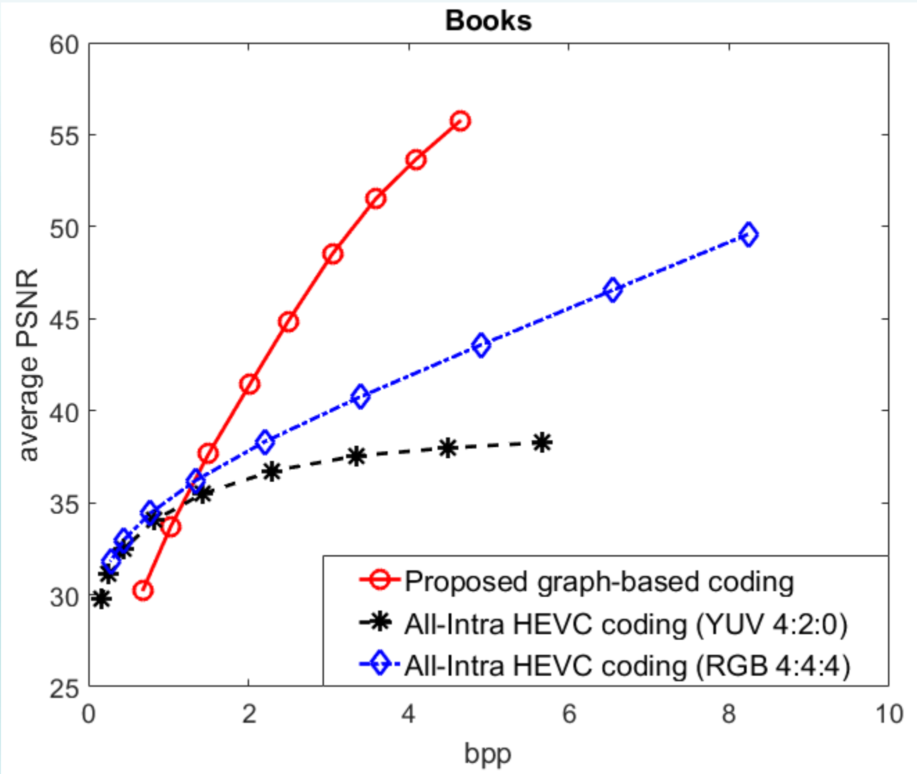


Pre-Demosiac Light Field Image Compression Using Graph Lifting Transform

- **Experimental Results**:

Dataset: EPFL light field image dataset
Baseline: All-intra HEVC coding in YUV4:2:0 and RGB 4:4:4





[1] Y.-H. Chao, G. Cheung, A. Ortega, "**Pre-Demosiac Light Field Image Compression Using Graph Lifting Transform**," *IEEE Int'l Conf. on Image Processing*, Beijing, China, September, 2017. (**Best student paper award**)

Outline

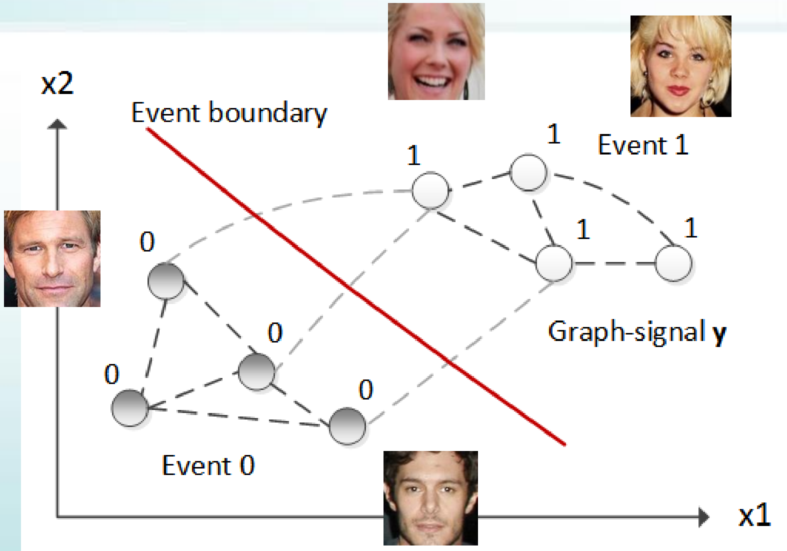
- Graph Signal Processing
  - Graph spectrum, GFT
- PWS Image Coding using GFT
- Prediction Residual Coding using GGFT
- Image Denoising using Graph Laplacian Regularizer
- Soft Decoding of JPEG Images w/ LERaG
- GSP for 3D Imaging
- Summary & Ongoing Work

Summary

- Graph Signal Processing (GSP)
  - Spectral analysis tools to process signals on graphs.
- PWS Image Compression
  - Graph Fourier Transform
  - Generalized GFT
  - Arithmetic Edge Coding
- Graph-signal Smoothness for Inverse Problems
  - Image denoising w/ graph Laplacian regularizer
  - New regularizer **LERaG** soft decoding of JPEG Images
- GSP for 3D Imaging
  - 3D point cloud compression, light field image compression

Other GSP Works:
Semi-Supervised Graph Classifier Learning

- **Binary Classifier**: given feature vector x_i of dimension K, compute $f(x_i) \in \{0,1\}$.
- **Classifier Learning**: given partial / noisy labels (x_i, y_i), train classifier $f(x_i)$.



example graph-based classifier

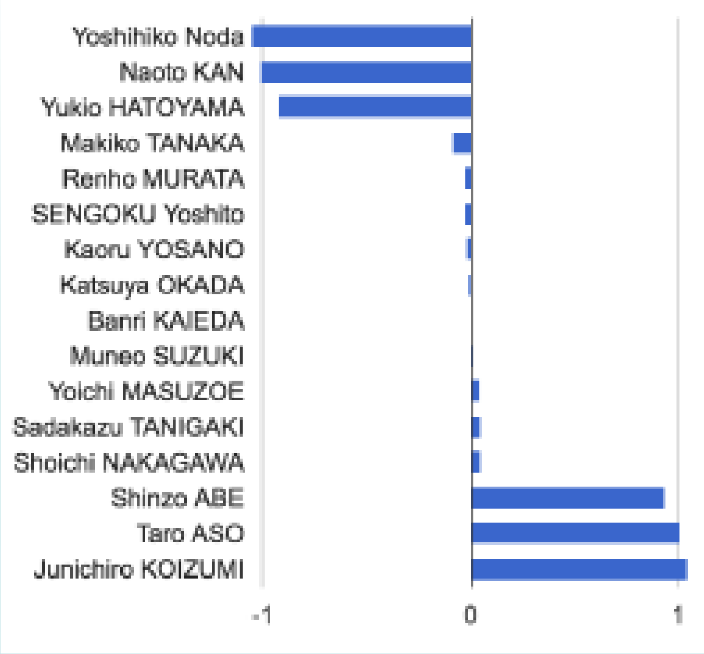
- **GSP Approach** [1]:
  1. Construct ***similarity graph*** with +/- edges.
  2. Pose MAP graph-signal restoration problem.
  3. Perturb graph Laplacian to ensure PSD.
  4. Solve num. stable MAP as sparse lin. system.

[1] Yu Mao, Gene Cheung, Chia-Wen Lin, Yusheng Ji, "**Image Classifier Learning from Noisy Labels via Generalized Graph Smoothness Priors**," *IEEE IVMSP Workshop*, Bordeaux, France, July 2016. (**Best student paper award**)

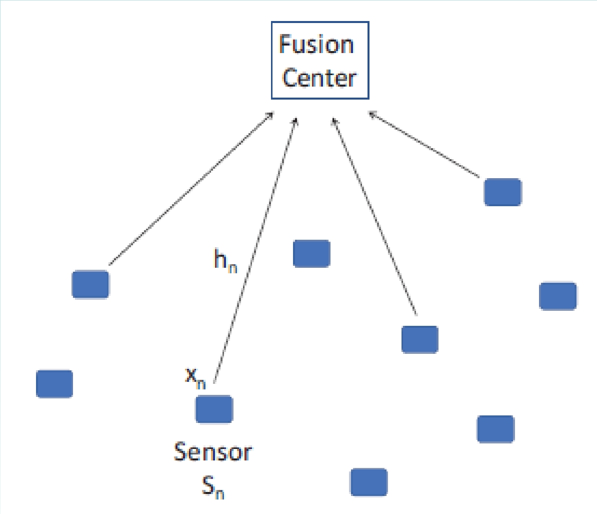
[2] G. Cheung, W.-T. Su, Y. Mao, C.-W. Lin, "Robust Semi-Supervised Graph Classifier Learning with Negative Edge Weights," submitted to *IEEE Transactions on Signal and Information Processing over Networks*, November 2016. (arXiv)

Other GSP Works

- Coding of spectral image [1], 3D point cloud w/ GFT.
- Coding of graph data w/ graph wavelets.
- Political leaning estimation [2].
- Wireless signal / power estimation [3].



[1] J. Zeng, G. Cheung, Y.-H. Chao, I. Blanes, J. Serra-Sagrista, A. Ortega, "**Hyperspectral Image Coding using Graph Wavelets**," *IEEE International Conference on Image Processing*, Beijing, China, September, 2017.



[2] B. Renoust et al., "**Estimation of Political Leanings via Graph-Signal Restoration**," *IEEE International Conference on Multimedia and Expo*, Hong Kong, China, July, 2017

[3] M. Kaneko, G. Cheung, W.-t. Su, C.-W. Lin, "**Graph-based Joint Signal / Power Restoration for Energy Harvesting Wireless Sensor Networks**," *IEEE Globecom*, Singapore, December, 2017.

Q&A

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