Signal Classification by Matching Node Connectivities[1]

Linh Lieu and *Naoki Saito*

Department of Mathematics
University of California
Davis, CA 95616 USA

IEEE Workshop on Statistical Signal Processing
Cardiff, Wales, UK
September 1, 2009

[1] Partially supported by NSF and ONR grants.

Outline

Outline

Given

- Training Data: $X = \{\mathbf{x}_1, \cdots, \mathbf{x}_{N_1}\} \subset \mathbb{R}^n$.
  Each data point $\mathbf{x}_j$ has a known class label $\in \{C_1, \cdots, C_K\}$.

Given

- Training Data: $X = \{\mathbf{x}_1, \cdots, \mathbf{x}_{N_1}\} \subset \mathbb{R}^n$.
  Each data point $\mathbf{x}_j$ has a known class label $\in \{C_1, \cdots, C_K\}$.
- Unlabeled (or Test) Data: $Y = \{\mathbf{y}_1, \cdots, \mathbf{y}_{N_2}\} \subset \mathbb{R}^n$.

Given

- Training Data: $X = \{\mathbf{x}_1, \cdots, \mathbf{x}_{N_1}\} \subset \mathbb{R}^n$.
  Each data point $\mathbf{x}_j$ has a known class label $\in \{C_1, \cdots, C_K\}$.
- Unlabeled (or Test) Data: $Y = \{\mathbf{y}_1, \cdots, \mathbf{y}_{N_2}\} \subset \mathbb{R}^n$.

Objective

Find a class label among $\{C_1, \cdots, C_K\}$ for each $\mathbf{y}_1, \cdots, \mathbf{y}_{N_2}$.

Our Approach

Following the Diffusion Framework:

Our Approach

Following the Diffusion Framework:

- Construct a similarity graph from the training data X, then expand the graph to the unlabeled data Y.

Our Approach

Following the Diffusion Framework:

- Construct a similarity graph from the training data X, then expand the graph to the unlabeled data Y.
- For each node, compute a histogram of node connectivity (distribution of its similarity to all the nodes), and let $\mathbf{h}_{\mathbf{x}_j}$ and $\mathbf{h}_{\mathbf{y}_k}$ denote the histogram corresponding to $\mathbf{x}_j \in X$ and $\mathbf{y}_k \in Y$, respectively.

Our Approach

Following the Diffusion Framework:

- Construct a similarity graph from the training data X, then expand the graph to the unlabeled data Y.
- For each node, compute a histogram of node connectivity (distribution of its similarity to all the nodes), and let $\mathbf{h}_{\mathbf{x}_j}$ and $\mathbf{h}_{\mathbf{y}_k}$ denote the histogram corresponding to $\mathbf{x}_j \in X$ and $\mathbf{y}_k \in Y$, respectively.
- Compare $\mathbf{h}_{\mathbf{x}_j}$ and $\mathbf{h}_{\mathbf{y}_k}$ using appropriate distance measure $\mathbf{d}(\cdot, \cdot)$.

Our Approach

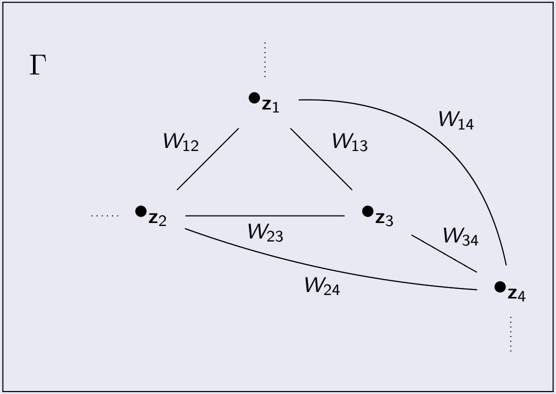
Following the Diffusion Framework:

- Construct a similarity graph from the training data X, then expand the graph to the unlabeled data Y.
- For each node, compute a histogram of node connectivity (distribution of its similarity to all the nodes), and let $\mathbf{h}_{\mathbf{x}_j}$ and $\mathbf{h}_{\mathbf{y}_k}$ denote the histogram corresponding to $\mathbf{x}_j \in X$ and $\mathbf{y}_k \in Y$, respectively.
- Compare $\mathbf{h}_{\mathbf{x}_j}$ and $\mathbf{h}_{\mathbf{y}_k}$ using appropriate distance measure $\mathbf{d}(\cdot,\cdot)$.
- Infer the label of $\mathbf{x}_{j^*}$ to $\mathbf{y}_k$ if

$$\mathbf{x}_{j^*} = \arg\min_{\mathbf{x}_j \in X} \mathbf{d}(\mathbf{h}_{\mathbf{x}_j}, \mathbf{h}_{\mathbf{y}_k})$$

Outline

First, build a connected Similarity Graph from the given data $Z = X \cup Y$.



Gaussian Weights: $W_{ij} \triangleq \mathrm{e}^{-\|\mathbf{z}_i - \mathbf{z}_j\|^2/\varepsilon^2}, \ \varepsilon > 0.$

Next, do Graph-Laplacian normalization to get the *diffusion* matrix:

Let D be the diagonal matrix:

$$D_{ii} \triangleq \sum_{k=1}^{N_1+N_2} \mathrm{e}^{-\|\mathbf{z}_i - \mathbf{z}_k\|^2/\varepsilon^2}, \; i = 1, 2, \cdots, N_1 + N_2.$$

where D_{ii} is the *degree* of node $\mathbf{z}_i$.

The *diffusion* matrix (size $(N_1 + N_2) \times (N_1 + N_2)$) is:

$$\boxed{P \triangleq D^{-1} W}$$

Properties of the diffusion matrix

Properties of the diffusion matrix

- P is non-negative and row-stochastic ($\sum_k P_{ik} = 1$).

Properties of the diffusion matrix

- P is non-negative and row-stochastic ($\sum_k P_{ik} = 1$).
- P represents a transition matrix of a Markov process on Γ.
  P_{ij} = probability of moving from $\mathbf{z}_i$ to $\mathbf{z}_j$ in one step.

Properties of the diffusion matrix

- P is non-negative and row-stochastic ($\sum_k P_{ik} = 1$).
- P represents a transition matrix of a Markov process on Γ.
  P_{ij} = probability of moving from $\mathbf{z}_i$ to $\mathbf{z}_j$ in one step.
- Spectrum of P: $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{N_1+N_2} \geq 0$.

Properties of the diffusion matrix

- P is non-negative and row-stochastic ($\sum_k P_{ik} = 1$).
- P represents a transition matrix of a Markov process on Γ.
  P_{ij} = probability of moving from $\mathbf{z}_i$ to $\mathbf{z}_j$ in one step.
- Spectrum of P: $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{N_1+N_2} \geq 0$.
- P has spectral decomposition:

$$P_{ij} = \sum_k \lambda_k \phi_k(i) \psi_k(j),$$

where

▷ $\{\phi_k\}$ and $\{\psi_k\}$ are (orthonormal) left and right eigenvectors,
▷ $\phi_k(i)$ = the ith entry of the vector ϕ_k.

Properties of the diffusion matrix

- P is non-negative and row-stochastic ($\sum_k P_{ik} = 1$).
- P represents a transition matrix of a Markov process on Γ. P_{ij} = probability of moving from $\mathbf{z}_i$ to $\mathbf{z}_j$ in one step.
- Spectrum of P: $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{N_1+N_2} \geq 0$.
- P has spectral decomposition:

$$P_{ij} = \sum_k \lambda_k \phi_k(i)\psi_k(j),$$

where
▷ $\{\phi_k\}$ and $\{\psi_k\}$ are (orthonormal) left and right eigenvectors,
▷ $\phi_k(i)$ = the ith entry of the vector ϕ_k.

- Markov process can be forwarded in time $t \in \mathbb{N}$ with $P^t_{ij} = \sum_k \lambda^t_k \phi_k(i)\psi_k(j)$ = prob. of moving from $\mathbf{z}_i$ to $\mathbf{z}_j$ in t steps.

Properties of the diffusion matrix

- P is non-negative and row-stochastic ($\sum_k P_{ik} = 1$).
- P represents a transition matrix of a Markov process on Γ. P_{ij} = probability of moving from $\mathbf{z}_i$ to $\mathbf{z}_j$ in one step.
- Spectrum of P: $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{N_1+N_2} \geq 0$.
- P has spectral decomposition:

$$P_{ij} = \sum_k \lambda_k \phi_k(i) \psi_k(j),$$

  where
  ▷ $\{\phi_k\}$ and $\{\psi_k\}$ are (orthonormal) left and right eigenvectors,
  ▷ $\phi_k(i)$ = the ith entry of the vector ϕ_k.
- Markov process can be forwarded in time $t \in \mathbb{N}$ with $P^t_{ij} = \sum_k \lambda^t_k \phi_k(i) \psi_k(j)$ = prob. of moving from $\mathbf{z}_i$ to $\mathbf{z}_j$ in t steps.
- Markov process has stationary distribution: $\boldsymbol{\pi} \triangleq \frac{D\mathbf{1}}{\mathbf{1}^T D\mathbf{1}}$

Properties of the diffusion matrix (cont.)

- The ith row $P_{i\cdot}$ can be viewed as a distribution of connectivity (degree) of $\mathbf{z}_i$ to all other nodes.

Properties of the diffusion matrix (cont.)

- The ith row $P_{i\cdot}$ can be viewed as a distribution of connectivity (degree) of $\mathbf{z}_i$ to all other nodes.

Definition of the diffusion distance

With $t \in \mathbb{N}$ given a priori, the *diffusion distance* $D_t(\mathbf{z}_i, \mathbf{z}_j)$ at time t is defined as:

$$\begin{aligned} D_t(\mathbf{z}_i, \mathbf{z}_j)^2 &\triangleq \left\| P^t_{i\cdot} - P^t_{j\cdot} \right\|^2_{L^2(X, \frac{1}{\pi})} \\ &= \sum_\ell \frac{\left(P^t_{i\ell} - P^t_{j\ell}\right)^2}{\pi(\ell)} \\ &= \sum_\ell \lambda_\ell^{2t} \left(\psi_\ell(i) - \psi_\ell(j)\right)^2 . \end{aligned} \tag{1}$$

Advantages of the diffusion distance $D_t(\mathbf{z}_i, \mathbf{z}_j)$

- preserves local neighborhood;
- measures the difference of how $\mathbf{z}_i$ and $\mathbf{z}_j$ are connected to all other nodes in Γ;
- takes into account all incidences relating $\mathbf{z}_i$ and $\mathbf{z}_j$;
- is robust to noise.

Approximation of the diffusion distance

Since eigenvalues λ_ℓ are decreasing to 0, diffusion distance can be approximated to a chosen accuracy $\tau > 0$:

$$D_t(\mathbf{z}_i, \mathbf{z}_j)^2 \approx \sum_{\ell=1}^{s(\tau,t)} \lambda_\ell^{2t} \left(\psi_\ell(i) - \psi_\ell(j)\right)^2,$$

for some $s(\tau, t) \in \mathbb{N}$.

Diffusion Map

- The Diffusion Map $\Psi_t : Z \to \mathbb{R}^{s(\tau,t)}$ is defined by

$$\Psi_t : \mathbf{z}_i \mapsto \left(\lambda_1^t \psi_1(i),\ \lambda_2^t \psi_2(i), \cdots,\ \lambda_{s(\tau,t)}^t \psi_{s(\tau,t)}(i)\right)^T.$$

Diffusion Map

- The Diffusion Map $\Psi_t : Z \to \mathbb{R}^{s(\tau,t)}$ is defined by

$$\Psi_t : \mathbf{z}_i \mapsto \left(\lambda_1^t \psi_1(i),\ \lambda_2^t \psi_2(i), \cdots,\ \lambda_{s(\tau,t)}^t \psi_{s(\tau,t)}(i) \right)^T .$$

- Ψ_t embeds Z into a low-dimensional *diffusion space*, $s(\tau, t) \ll n$.

Diffusion Map

- The Diffusion Map $\Psi_t : Z \to \mathbb{R}^{s(\tau,t)}$ is defined by

$$\Psi_t : \mathbf{z}_i \mapsto \left(\lambda_1^t \psi_1(i),\ \lambda_2^t \psi_2(i), \cdots,\ \lambda_{s(\tau,t)}^t \psi_{s(\tau,t)}(i)\right)^T.$$

- Ψ_t embeds Z into a low-dimensional *diffusion space*, $s(\tau, t) \ll n$.
- $D_t(\mathbf{z}_i, \mathbf{z}_j) \approx \|\Psi_t(\mathbf{z}_i) - \Psi_t(\mathbf{z}_j)\|$, diffusion distance is approximated by Euclidean distance within the diffusion space.

In practice for signal classification problems

We do not compute diffusion maps on $Z = X \cup Y$.

- ▷ Compute the similarity graph Γ only from the training data X.
  $\Rightarrow$ diffusion maps are defined only for X, $\Psi_t : X \to \mathbb{R}^{s(\tau,t)}$.
- ▷ Extend Ψ_t to Y (out-of-sample extension) using
  - Geometric harmonics multiscale extension scheme (Lafon-Keller-Coifman, 2006); or
  - Nyström extension (Fowlkes-Belongie-Chung-Malik, 2004).

  $\Rightarrow$ After which $\Psi_t : X \cup Y \to \mathbb{R}^{s(\tau,t)}$.
- ▷ Diffusion distance between $\mathbf{x}_i \in X$ and $\mathbf{y}_j \in Y$ is approximately $D_t(\mathbf{x}_i, \mathbf{y}_j) \approx \|\Psi_t(\mathbf{x}_i) - \Psi_t(\mathbf{y}_j)\|$.

Outline

Our Proposed Method: Node Connectivities Matching

- A modification to the diffusion distance approach.

Our Proposed Method: Node Connectivities Matching

- A modification to the diffusion distance approach.
- No computation of eigenvalues/eigenvectors.

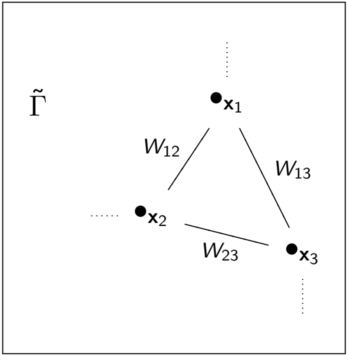
Our Proposed Method: Node Connectivities Matching

- A modification to the diffusion distance approach.
- No computation of eigenvalues/eigenvectors.
- Bypass the out-of-sample extension step, hence avoid error admitted during the extension process.

Our Proposed Method: Node Connectivities Matching

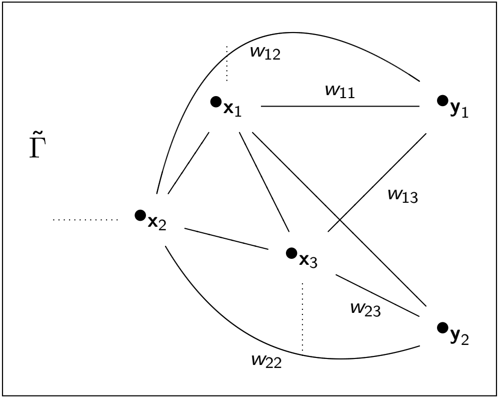
- A modification to the diffusion distance approach.
- No computation of eigenvalues/eigenvectors.
- Bypass the out-of-sample extension step, hence avoid error admitted during the extension process.
- Still close to the diffusion distance, hence inherits nice local-neighborhood preserving property from the diffusion distance.

Set up a connected Similarity Graph $\tilde{\Gamma}$ from training data X.



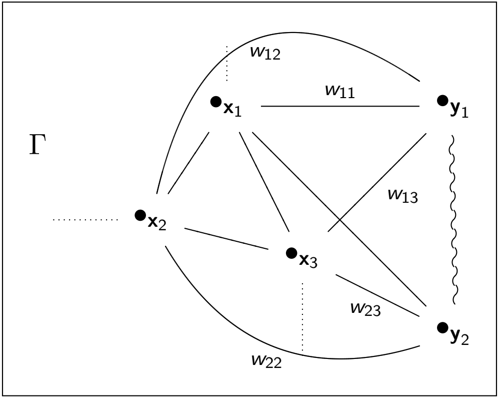
Let $\mathbf{h}_{\mathbf{x}_j} \triangleq \frac{1}{\sum_{\ell=1}^{N_1} W_{j\ell}} (W_{j1}, \cdots, W_{jN_1}) \in \mathbb{R}^{1 \times N_1}$, the degree distribution or histogram of connectivities of node $\mathbf{x}_j$ to all other nodes in $\tilde{\Gamma}$.

Add nodes corresponding to unlabeled points in Y. Only add edges connecting between X and Y with weights $w_{j\ell} = \mathrm{e}^{-\|\mathbf{y}_j - \mathbf{x}_\ell\|^2/\varepsilon^2}$.



Let $\mathbf{h}_{\mathbf{y}_j} \triangleq \frac{1}{\sum_{\ell=1}^{N_1} w_{j\ell}} (w_{j1}, \cdots, w_{jN_1}) \in \mathbb{R}^{1 \times N_1}$, the degree distribution or histogram of connectivities of node $\mathbf{y}_j$ to all $\mathbf{x}_\ell$ nodes in $\tilde{\Gamma}$.

$\tilde{\Gamma}$ differs from the original fully connected similarity graph Γ on $Z = X \cup Y$ only in the absence of the edges $(\mathbf{y}_j, \mathbf{y}_k)$.



Matching Node Connectivities

Discriminate the histogram $\mathbf{h}_{\mathbf{y}_j}$ from $\mathbf{h}_{\mathbf{x}_k}$ using various measures:

- L^2 measure: $L^2(\mathbf{h}_{\mathbf{y}_j}, \mathbf{h}_{\mathbf{x}_k}) = \sqrt{\sum_{\ell=1}^{N_1} |\mathbf{h}_{\mathbf{y}_j}(\ell) - \mathbf{h}_{\mathbf{x}_k}(\ell)|^2}$.
- Jeffreys divergence:
  $d_J(\mathbf{h}_{\mathbf{y}_j}, \mathbf{h}_{\mathbf{x}_k}) = \sum_{\ell=1}^{N_1} \left(\mathbf{h}_{\mathbf{y}_j}(\ell) \log \frac{\mathbf{h}_{\mathbf{y}_j}(\ell)}{\mathbf{h}_{\mathbf{x}_k}(\ell)} + \mathbf{h}_{\mathbf{x}_k}(\ell) \log \frac{\mathbf{h}_{\mathbf{x}_k}(\ell)}{\mathbf{h}_{\mathbf{y}_j}(\ell)} \right)$.
- Hellinger distance: $d_H(\mathbf{h}_{\mathbf{y}_j}, \mathbf{h}_{\mathbf{x}_k}) = \sum_{\ell=1}^{N_1} \left(\sqrt{\mathbf{h}_{\mathbf{y}_j}(\ell)} - \sqrt{\mathbf{h}_{\mathbf{x}_k}(\ell)} \right)^2$.
- χ^2 Statistics: $\chi^2(\mathbf{h}_{\mathbf{y}_j}, \mathbf{h}_{\mathbf{x}_k}) = \sum_{\ell=1}^{N_1} \frac{\left(\mathbf{h}_{\mathbf{y}_j}(\ell) - m(\ell)\right)^2}{m(\ell)}$,
  where $m(\ell) = \frac{1}{2}\left(\mathbf{h}_{\mathbf{y}_j}(\ell) + \mathbf{h}_{\mathbf{x}_k}(\ell)\right)$.
- Earth Mover's Distance See defn of EMD.

Outline

Triangular Waveform Classification

Triangular Waveforms

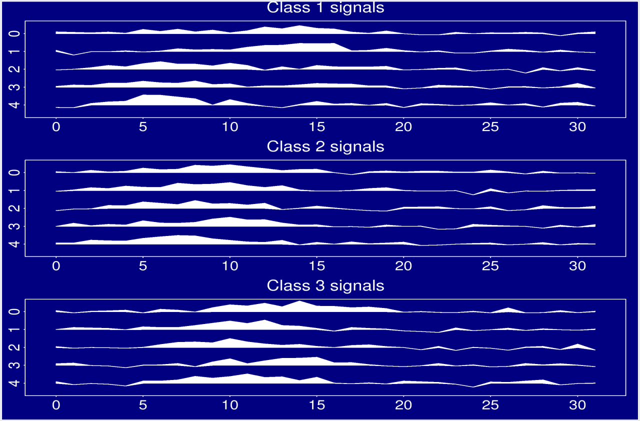


Figure: Five samples of three triangular waveform classes.

Triangular Waveform Classification

Triangular Waveform Data Generation

Three classes of signals generated via:

- $x^{(1)}(j) = uh_1(j) + (1-u)h_2(j) + \epsilon(j)$.
- $x^{(2)}(j) = uh_1(j) + (1-u)h_3(j) + \epsilon(j)$.
- $x^{(3)}(j) = uh_2(j) + (1-u)h_3(j) + \epsilon(j)$.

where

- $j = 1, \cdots, 32$.
- $h_1(j) = \max\{6 - |j-7|,\ 0\};\ \ h_2 = h_1(j-8);\ \ h_3(j) = h_1(j-4)$.
- u is a uniform random variable on interval $(0,1)$.
- ϵ is a standard normal variate.

Triangular Waveform Classification

Experimental Procedure

- Generate 100 training signals/class and 1000 test signals/class.
- Repeat the procedure 10 times to get average misclassification rates.

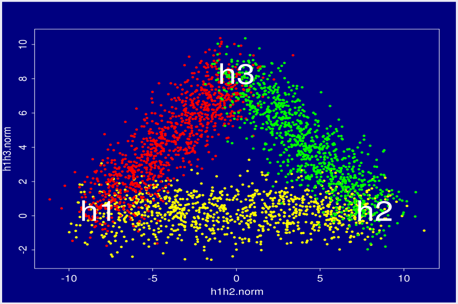


Figure: A 2D projection of the triangular waveform dataset.

Triangular Waveform Classification

Results (The *Bayes* rate is $\sim 14\%$)

Misclassification rates (average over 10 simulations)

NCM in	Error rate (%)
L^2 Distance	20.07
Jeffreys Divergence	19.47
Hellinger Distance	19.45
χ^2 Statistics	19.43
EMD	**16.43**

Classification by Nearest Neighbor Method

	Error rate (%)
Diff Maps extended by GHME:	19.21
Diff Maps computed on $Z = X \cup Y$	18.05
No Diff Maps (the original coordinates)	21.21

Hyperspectral Images of Natural Scenes

Each pixel is a vector of 43 reflectance values at various wavelengths.

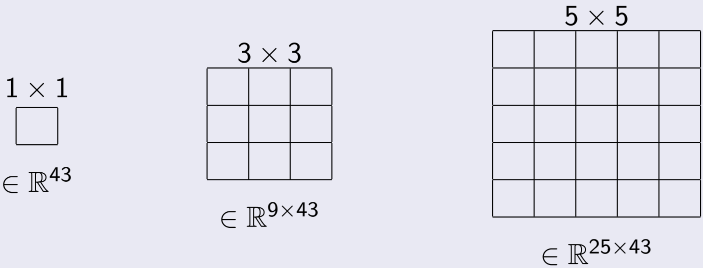
Reference:

D. L. Ruderman, *Statistics of cone responses to natural images: implications for visual coding*, J. Opt. Soc. Am., vol. 15, no. 8, pp. 2036–2045, August 1998.

Recognition of Pixel Type

Data preparation

Extract a window around each pixel

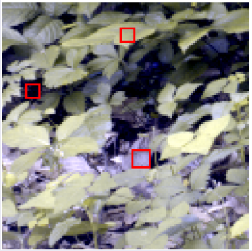


$\Longrightarrow$ the data point associated with a pixel is $\mathbf{x} \in \mathbb{R}^{43}$, $\mathbb{R}^{9\times 43}$, or $\mathbb{R}^{25\times 43}$.

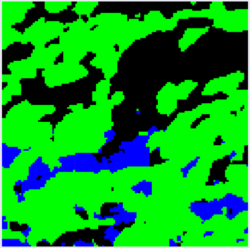
Recognition of Pixel Type

Recognition of pixel types from given samples

Three selected regions input as seeds



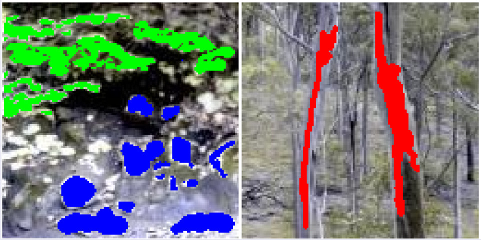
Recognition result



Quantitatively Controlled Recognition of Pixel Type

Data Preparation

Hand segment regions of leaves, rocks, trunks from different images.



Green: leaves; Blue: rocks; Red: trunks.

Quantitatively Controlled Recognition of Pixel Type

Experimental Procedure

- Three-class recognition problem: Leaf, Rock, and Trunk pixels.
- Randomly select $\approx$ 400 pixels per class for training and $\approx$ 1200 pixels per class for test.
- Training and test data come from different images.
- Repeat procedure 20 times to get average misclassification rates.

Quantitatively Controlled Recognition of Pixel Type

Results

Classification via NCM

Misclassification rates (average over 20 simulations)

1×1		3×3		5×5	
Measure	Error (%)	Measure	Error (%)	Measure	Error (%)
L^2	42.63	L^2	**20.00**	L^2	20.06
Jeffreys	29.48	Jeffreys	23.77	Jeffreys	21.76
Hellinger	27.13	Hellinger	21.92	Hellinger	**20.00**
χ^2	28.89	χ^2	27.39	χ^2	20.85
EMD	46.03	EMD	30.76	EMD	22.93

Classification via nearest neighbor in

Measure	Error (%)	Measure	Error (%)	Measure	Error (%)
L^2 dist.	31.83	L^2 dist.	31.26	L^2 dist.	30.05
Diff. dist.	24.85	Diff. dist.	57.23	Diff. dist.	50.92

Outline

Summary

Node Connectivity Matching (NCM)

- Is derived from diffusion distance;
- Bypasses computation of eigenvalues/eigenvectors of diffusion operator;
- Avoids out-of-sample extension.

$\Longrightarrow$ Admits less error !

References for Diffusion Distance and Diffusion Framework:

- R. R. Coifman, S. Lafon, *Diffusion maps*, Appl. Comput. Harmon. Anal., **21**:5–30, July 2006.
- S. Lafon, Y. Keller, R.R. Coifman, *Data fusion and multicue data matching by diffusion maps*, IEEE Trans. Pattern Anal. Machine Intell., **28**(11):1784–1797, 2006.
- C. Fowlkes, S. Belongie, F. Chung, J. Malik, *Spectral grouping using the Nyström method*, IEEE Trans. Pattern Anal. Machine Intell., **26**(2):214–225, 2004.
- L. Lieu, N. Saito, *High-dimensional pattern recognition using low-dimensional embedding and Earth Mover's Distance*, submitted for publication, 2009. Available at `http://www.math.ucdavis.edu/~saito/publications/`.

Definition of Earth Mover's Distance (EMD)

Suppose $\mathbf{p} = (p_1, \cdots, p_{n_1})$ and $\mathbf{q} = (q_1, \cdots, q_{n_2})$ are two histograms or two discrete distributions. The *Earth Mover's Distance* (EMD) is defined by

$$\mathrm{EMD}(\mathbf{p}, \mathbf{q}) \triangleq \frac{\sum_{i,j} g_{ij} d_{ij}}{\sum_{i,j} g_{ij}}$$

where

- $d_{ij}, \; i = 1, \cdots, n_1, \; j = 1, \cdots, n_2$: ground distance; the dissimilarity between bins i and j; the cost of moving one unit of feature in the feature space between the ith and jth feature,
- $g_{ij} \geq 0, \; i = 1, \cdots, n_1, \; j = 1, \cdots, n_2$: the optimal flow between two histograms that minimizes the total cost $\sum_{i,j} g_{ij} d_{ij}$, subject to the following constraints
  - ▷ $\sum_i g_{ij} \leq q_j$,
  - ▷ $\sum_j g_{ij} \leq p_i$,
  - ▷ $\sum_{ij} g_{ij} = \min\{\sum_i p_i, \sum_j q_j\}$

Y. Rubner, C. Tomasi, L. J. Guibas, *The Earth Mover's Distance as a Metric for Image Retrieval*, International Journal of Computer Vision, **40**(2): 99–121, 2000.

▸ back to prev slide