Image Masking Schemes for Local Manifold Learning Methods

Marco F. Duarte

UMASS AMHERST

Joint work with Hamid Dadkhahi

IEEE ICASSP - April 23 2015

Manifold Learning

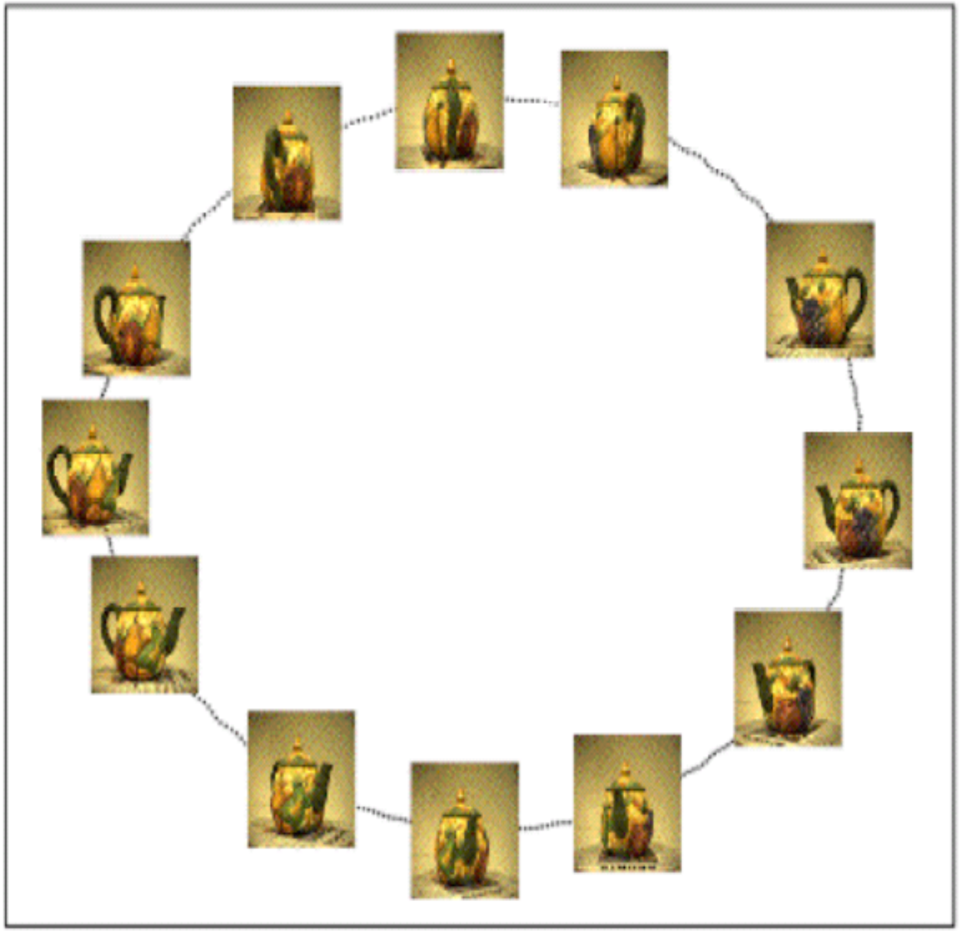
- Given training points in $\mathbb{R}^N$, learn the mapping to the underlying K-dimensional articulation manifold

- Exploit ***local geometry*** to capture parameter differences by embedding distances

- ISOMAP, LLE, HLLE, …

- Ex: images of
  rotating teapot

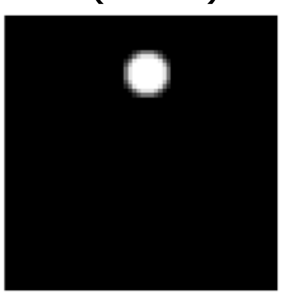
  articulation space
  = circle



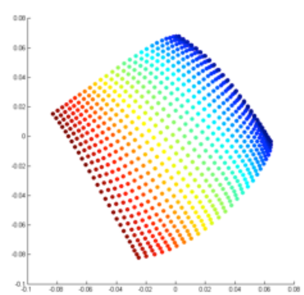
Compressive Manifold Learning

- Given training points in $\mathbb{R}^N$, learn the mapping to the underlying K-dimensional articulation manifold
- Isomap algorithm approximates geodesic distances using ℓ_2 distances between neighboring points
- Random measurements preserve these distances
- **Theorem**: [Hegde, Wakin, Baraniuk 2008] If $M = \mathcal{O}(K \log(N)/\delta^2)$, then the Isomap residual variance in the projected domain is bounded by the additive error factor $R_\Phi < R + C\delta$

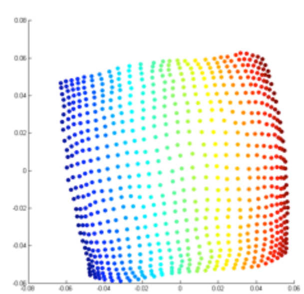
translating disk manifold (K=2)



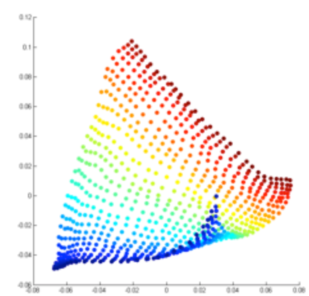
$N = 4096$ (Full Data)



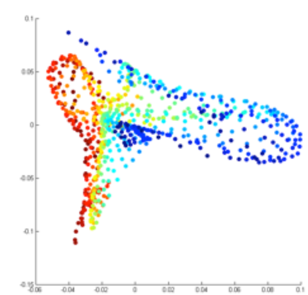
$M = 100$



$M = 50$



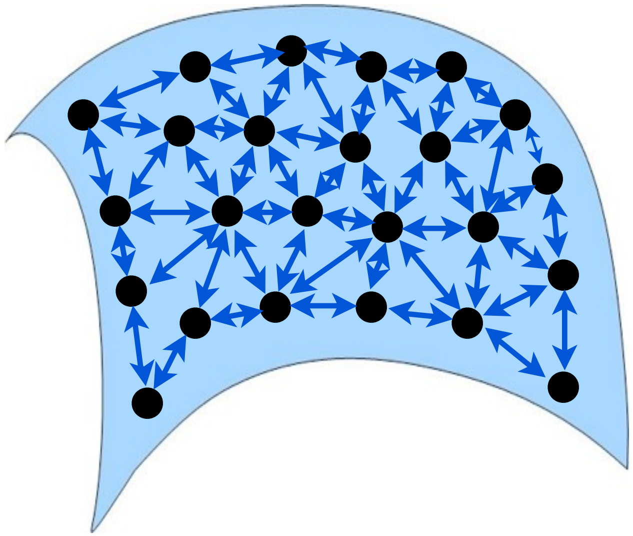
$M = 25$



Custom Projection Operators

- Goal of Dimensionality Reduction: To ***preserve distances*** between points in the manifold, i.e., for $x_1, x_2 \in \mathcal{M}$

$$(1-\epsilon) \leq \frac{\|\Phi x_1 - \Phi x_2\|_2^2}{\|x_1 - x_2\|_2^2} \leq (1+\epsilon)$$



- Collect pairwise differences into set of ***secant vectors***

$$\mathcal{S} = \left\{ \frac{x_1 - x_2}{\|x_1 - x_2\|_2} \ : \ x_1, x_2 \in \mathcal{M} \right\}$$

- Search for projection that preserves norms of secants:

$$\|\Phi s_i\|_2^2 \approx 1 \text{ for all } s_i \in \mathcal{S}$$

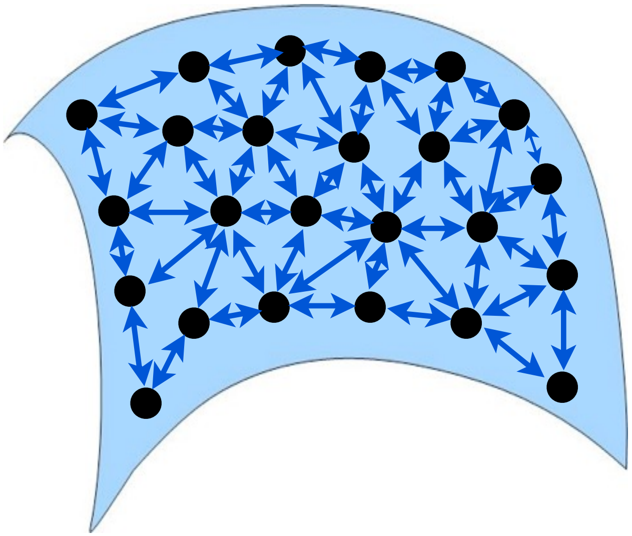
$$s_i^T \Phi^T \Phi s_i \approx 1$$

[Hegde, Sankaranarayanan, Baraniuk 2012]

Custom Projection Operators

- Usual approach: ***Principal Component Analysis*** (PCA)
- Collect all secants into a matrix:

$$S = [s_1 \ s_2 \ \ldots \ s_L], \ s_i \in \mathcal{S}$$



- Perform eigenvalue decomposition on S:

$$S = U \Sigma V^T$$

- Select top eigenvectors as projections

$$\Phi = (U_{1:M})^T$$

- PCA minimizes the average squared distortion over secants, but ***can distort individual secants arbitrarily*** and therefore warp manifold structure

[Hegde, Sankaranarayanan, Baraniuk 2012]

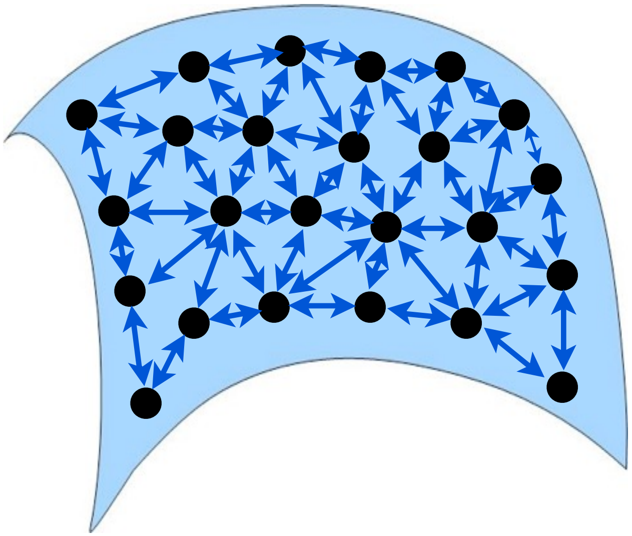
Custom Projection Operators: NuMax

- For target distortion ϵ, find matrix Φ featuring the ***smallest number of rows*** that yields

$$|s_i^T \Phi^T \Phi s_i - 1| \leq \epsilon \text{ for each } s_i \in \mathcal{S}$$

- This is equivalent to minimizing the ***rank*** of the matrix $P = \Phi^T \Phi$ such that

$$|s_i^T P s_i - 1| \leq \epsilon \text{ for each } s_i \in \mathcal{S}$$

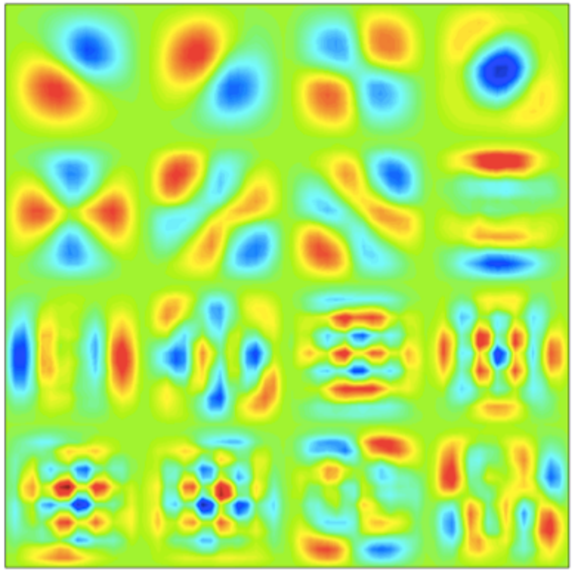


- Use ***nuclear norm*** as proxy for rank to obtain computationally efficient approach
- Improves over random projections since matrix Φ is ***specifically tailored*** to manifold observed
- May be difficult to link target distortion ϵ to matrix rank/ number of rows Φ

[Hegde, Sankaranarayanan, Baraniuk 2012]

Issues with Randomness and NuMax

- Projections matrices have entries with arbitrary values
- Physics of sensing process, hardware devices ***restrict types of projections*** we can obtain
- Example: Low-power imaging for computational eyeglasses
- Low-power imaging sensor allows for ***individual selection*** of pixels to record
- Power consumption ***proportional*** to number of pixels sampled
- Random projections/NuMax involve half/all pixels and ***do not enable power savings***
- How to derive constrained projection matrices that involve only few pixels?

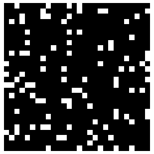


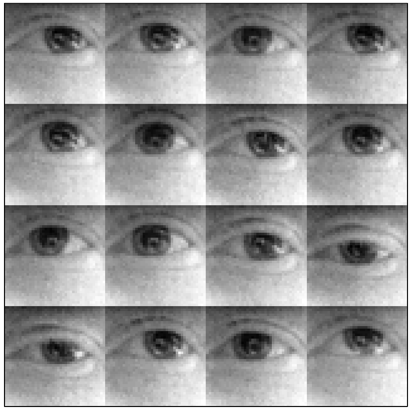
[Mayberry, Hu, Marlin, Salthouse, Ganesan 2014]

Masking Strategies for Manifold Data

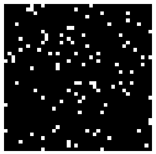
- Select ***only a subset*** of the pixels of size M that minimizes distortion to manifold structure
- Emulate strategies for projection design into mask design
- ***Random Masking***:
  Pick M pixels uniformly at random across image

$N = 28 \times 28 = 784$





$N = 40 \times 40 = 1600$



$M = 100$
pixels

Masking Strategies for Manifold Data

- Select ***only a subset*** of the pixels of size M that minimizes distortion to manifold structure

- Emulate strategies for projection design into mask design

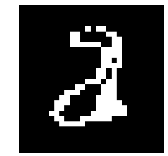
- ***Principal coordinate analysis***: Pick M coordinates that maximize variance among secants

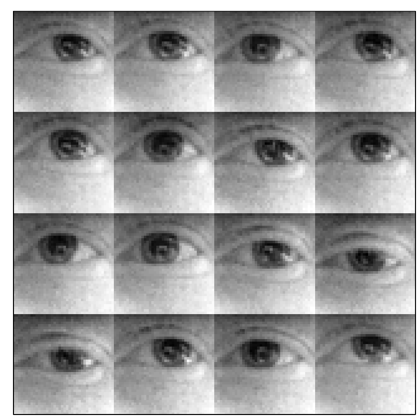
$$\omega_i = \arg \max_{i \in [N]} \sum_{\ell=1}^{L} \left(s_\ell(i) - \bar{s}(i)\right)^2$$

$$\text{subject to} \quad \omega_i \neq \omega_j \quad \forall\, j < i,$$

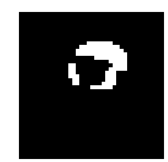
[Dadkhahi and Duarte 2014]

$N = 28 \times 28 = 784$





$N = 40 \times 40 = 1600$



$M = 100$ pixels

Masking Strategies for Manifold Data

- Select ***only a subset*** of the pixels of size M that minimizes distortion to manifold structure
- Emulate strategies for projection design into mask design

- ***Adaptation of NuMax***:
- Define secants from k-nearest neighbor graph:

$$\mathcal{S}_k(x_i) := \{x_{j_1} - x_i : x_{j_1} \in \mathcal{C}_k(x_i)\}$$

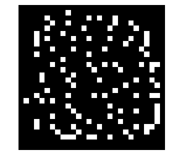
- Pick $M \times N$ masking matrix Φ (row submatrix of I) to minimize secant norm distortion after scaling:

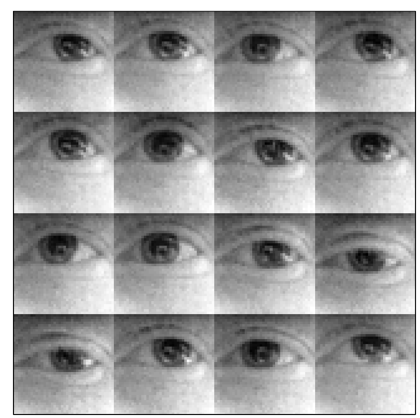
$$\left| s_i^T \Phi^T \Phi s_i - \frac{M}{N} \right| \leq \epsilon \text{ for each } s_i \in \mathcal{S}$$

- Combinatorial integer program replaced by greedy approximation

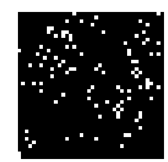
[Dadkhahi and Duarte 2014]

$N = 28 \times 28 = 784$





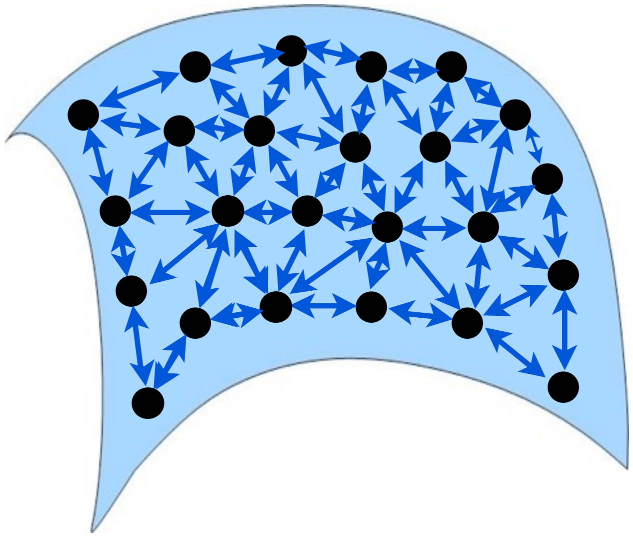
$N = 40 \times 40 = 1600$



$M = 100$ pixels

Isomap vs. Locally Linear Embedding

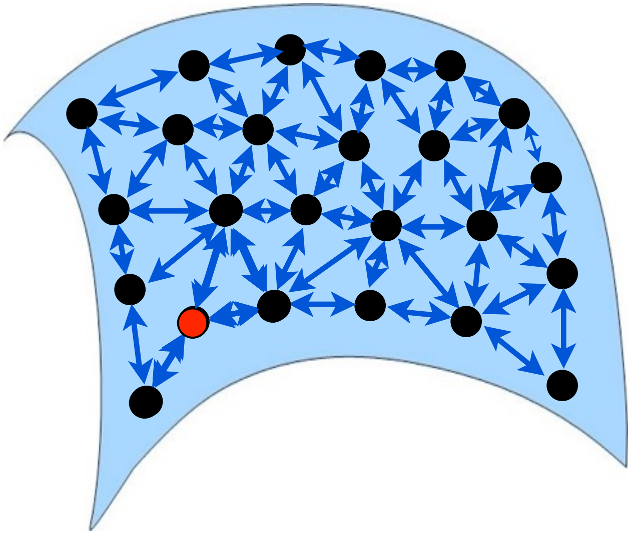
- While Isomap employs distances between neighbors when designing the embedding, ***Locally Linear Embedding*** (LLE) employs ***additional local geometry***, representing each vector as a weighted linear combination of its neighbors



- In particular, Isomap embedding is sensitive to scaling of the point clouds, while LLE isn't

Isomap vs. Locally Linear Embedding

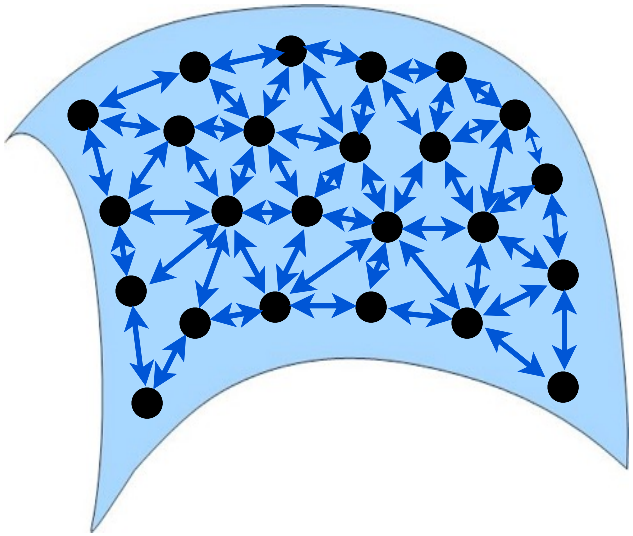
- While Isomap employs distances between neighbors when designing the embedding, ***Locally Linear Embedding*** (LLE) employs ***additional local geometry***, representing each vector as a weighted linear combination of its neighbors



- In particular, Isomap embedding is sensitive to scaling of the point clouds, while LLE isn't

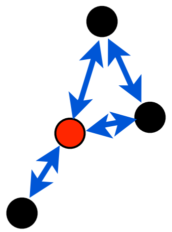
Isomap vs. Locally Linear Embedding

- While Isomap employs distances between neighbors when designing the embedding, ***Locally Linear Embedding*** (LLE) employs ***additional local geometry***, representing each vector as a weighted linear combination of its neighbors



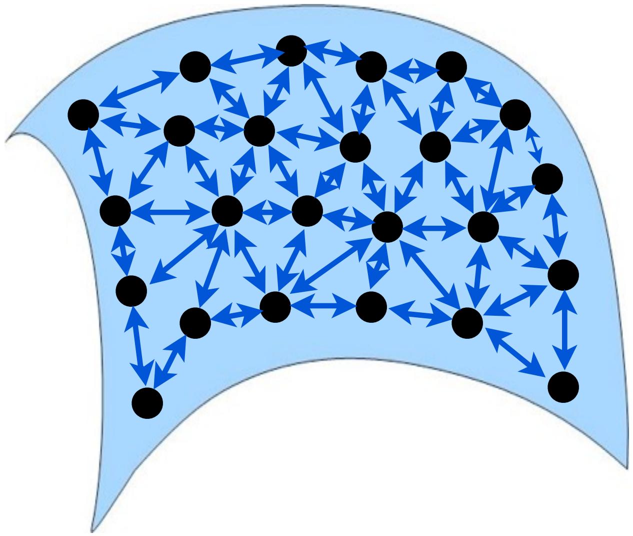
- In particular, Isomap embedding is sensitive to scaling of the point clouds, while LLE isn't

- To preserve this additional local information, we ***expand*** the set of secants to include distances between neighbors of each point

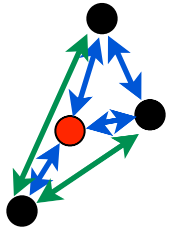


Isomap vs. Locally Linear Embedding

- While Isomap employs distances between neighbors when designing the embedding, ***Locally Linear Embedding*** (LLE) employs ***additional local geometry***, representing each vector as a weighted linear combination of its neighbors



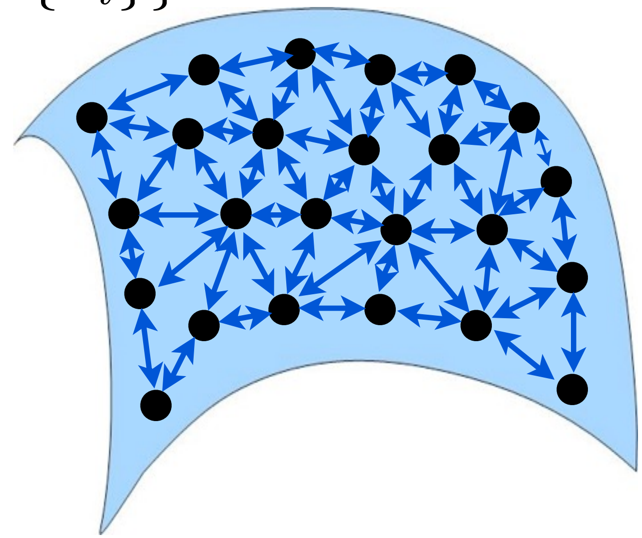
- In particular, Isomap embedding is sensitive to scaling of the point clouds, while LLE isn't
- To preserve this additional local information, we ***expand*** the set of secants to include distances between neighbors of each point



Manifold-Aware Pixel Selection for LLE

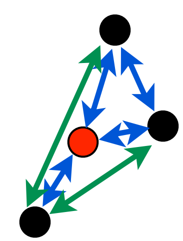
- Expand the set of secants considered:

$$\mathcal{S}_k(x_i) := \{x_{j_1} - x_{j_2} : x_{j_1}, x_{j_2} \in \mathcal{C}_k(x_i) \cup \{x_i\}\}$$



- Compute squared norms of secants in $\mathcal{S}_k(x_i)$ obtained from the original images and from masked images; collect into "norm" vectors α_i, β_i
- Choose mask that maximizes sum of cosine similarities between original and masked "norm" vectors:

$$S(\Omega) = \sum_i \mathrm{sim}(\alpha_i, \beta_i) := \sum_i \frac{\langle \alpha_i, \beta_i \rangle}{\|\alpha_i\|_2 \|\beta_i\|_2}$$



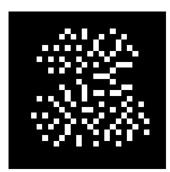
- Replace combinatorial optimization by greedy forward selection algorithm
- Cosine similarity is ***invariant to (local) scaling*** of point cloud

Manifold-Aware Pixel Selection for LLE

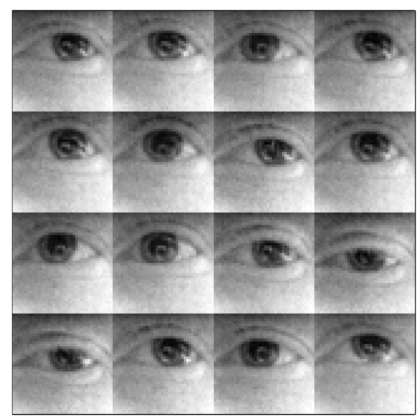
Algorithm 2 Manifold-Aware Pixel Selection for LLE (MAPS-LLE)

Inputs: neighborhood clique secant array B, masking size m
Outputs: masking index set Ω
Initialize: $\Omega \leftarrow \{\}$
$\alpha \leftarrow \sum_{j \in [d]} B(:, j, :)$ {Compute matrix of squared secant norms.}
for $i = 1 \rightarrow m$ **do**
  $\theta \leftarrow \sum_{j \in \Omega} B(:, j, :)$ {Compute matrix of squared masked secant norms for current masking set Ω.}
  for $j = 1 \rightarrow |\Omega^C|$ **do**
    $\beta \leftarrow \theta + B(:, j, :)$ {Update squared masked secant norms when $\Omega^C(j)$ is added to mask Ω.}
    $\lambda(j) \leftarrow \sum_{t \in [n]} \frac{\langle \alpha(:,t), \beta(:,t) \rangle}{||\alpha(:,t)||_2 ||\beta(:,t)||_2}$ {Compute cosine similarity measure for updated mask.}
  end for
  $\omega \leftarrow \arg\max_{j \in [|\Omega^C|]} \lambda(j)$ {Find new mask element that maximizes cosine similarity.}
  $\Omega \leftarrow \Omega \cup \{\Omega^C(\omega)\}$ {Add selected dimension to the masking index set.}
end for

$N = 28 \times 28 = 784$



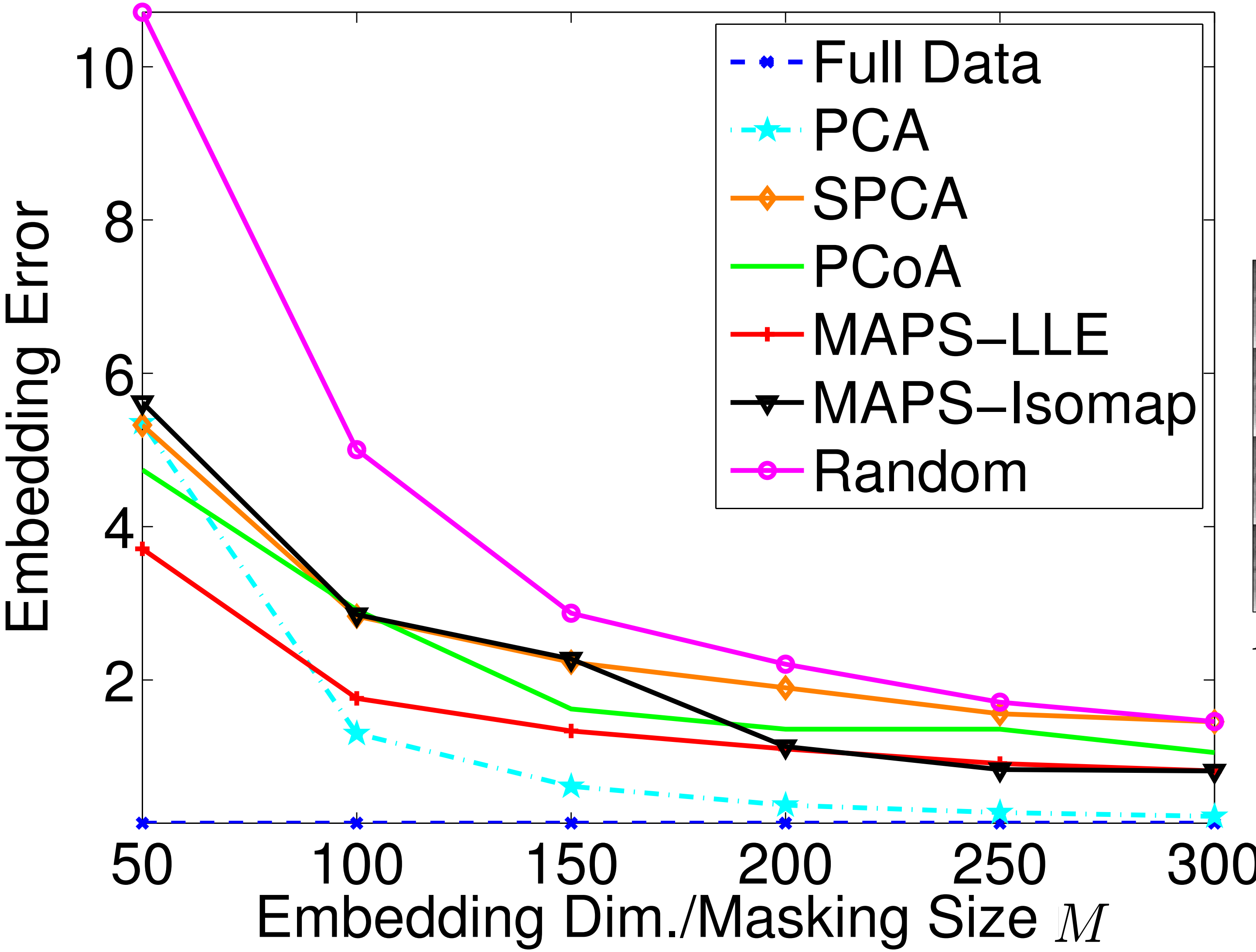
$M = 100$ pixels

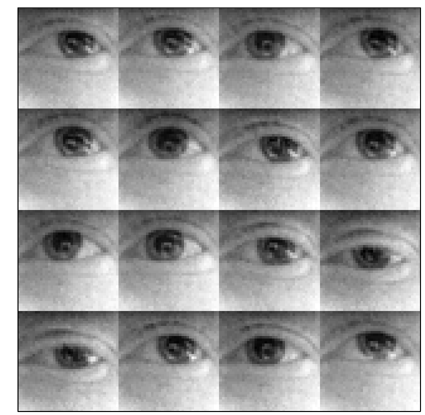


$N = 40 \times 40 = 1600$

$M = 100$ pixels

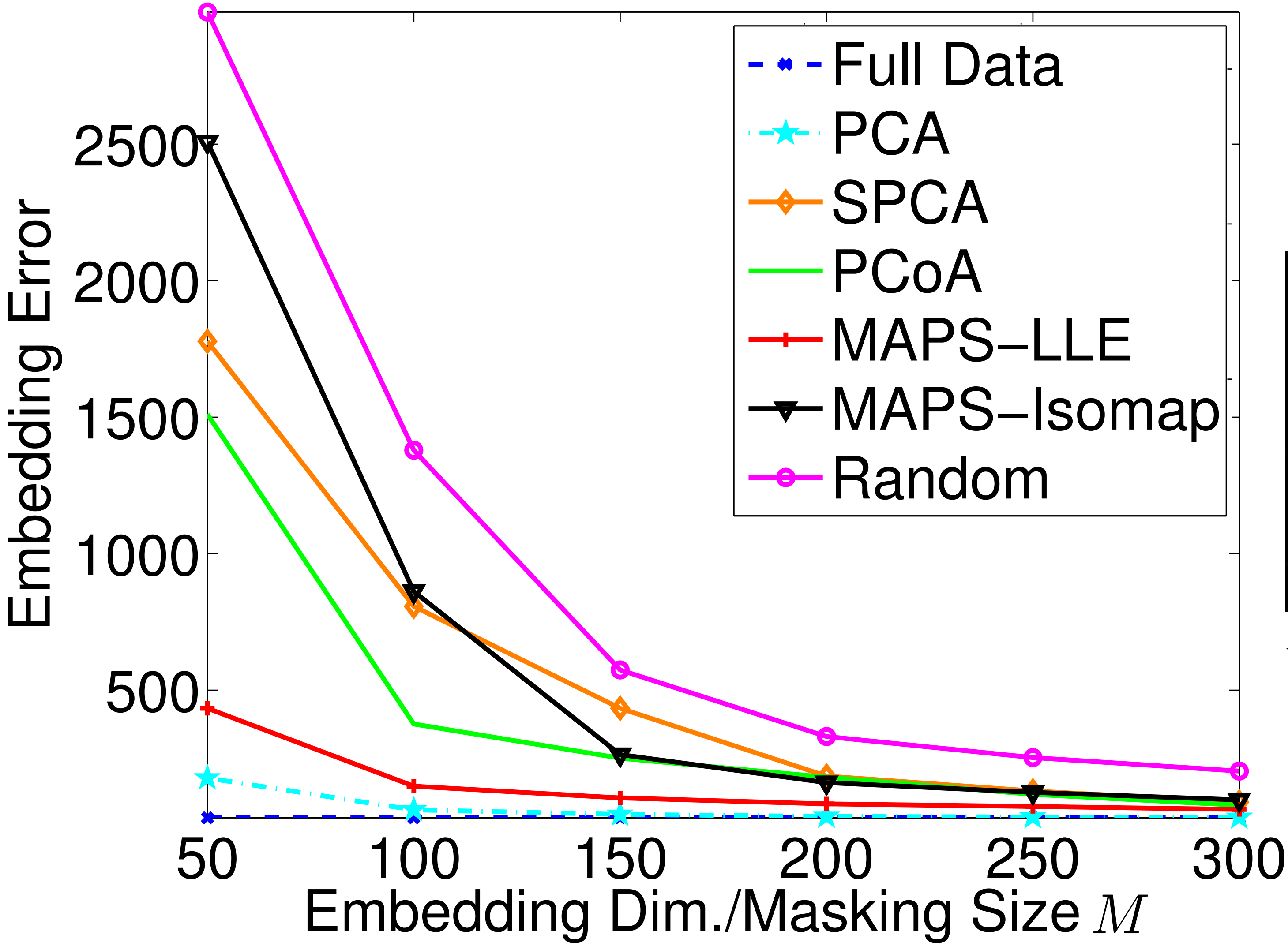
Performance Analysis: LLE Embedding Error





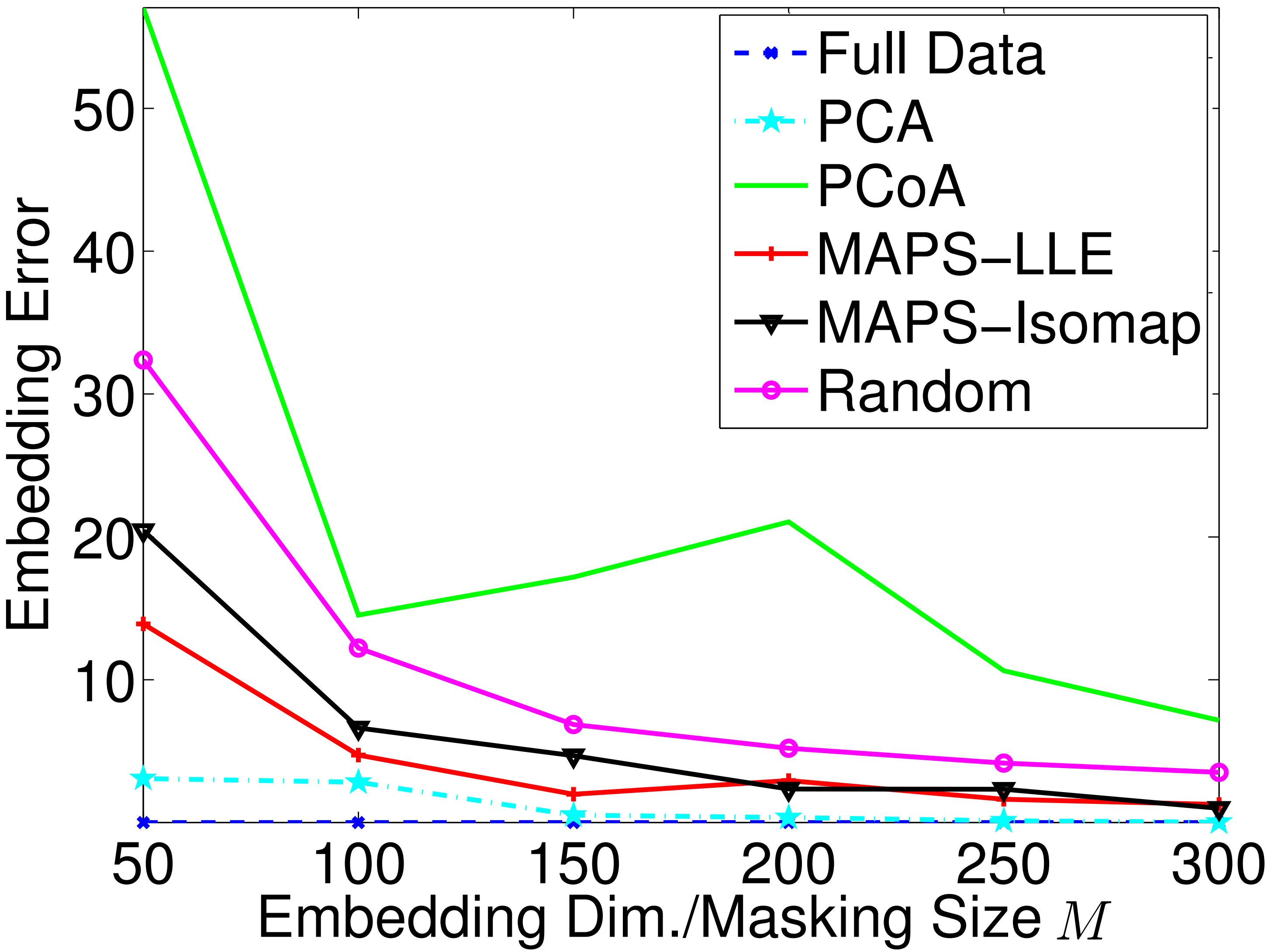
$N = 40 \times 40 = 1600$

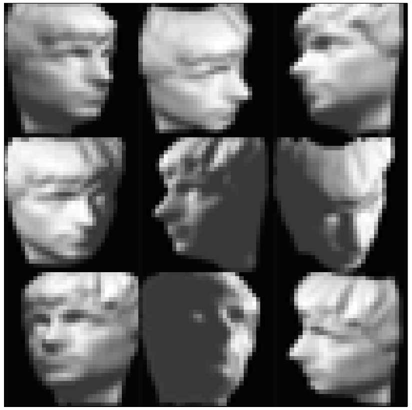
Performance Analysis: LLE Embedding Error



$N = 28 \times 28 = 784$

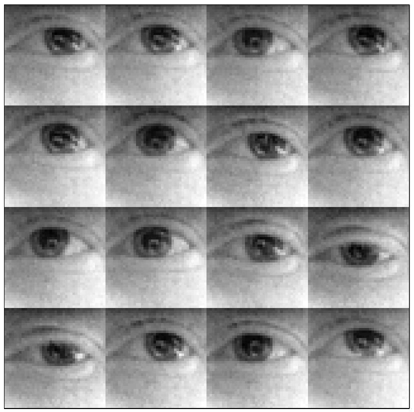
Performance Analysis: LLE Embedding Error





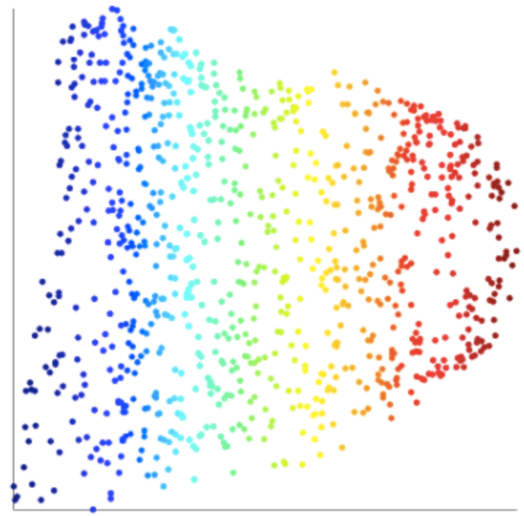
$N = 32 \times 32 = 1024$

Performance Analysis: 2-D LLE

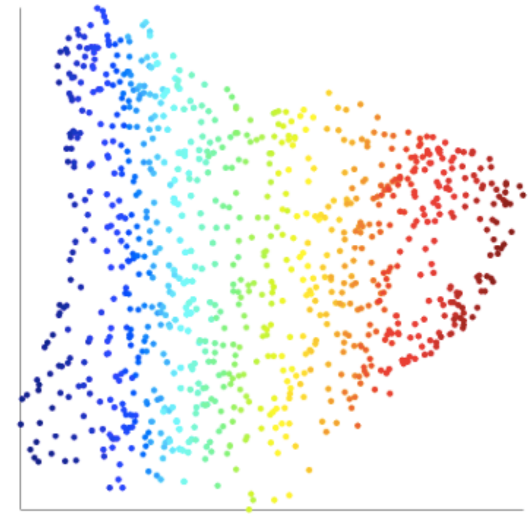


$N = 40 \times 40 = 1600$

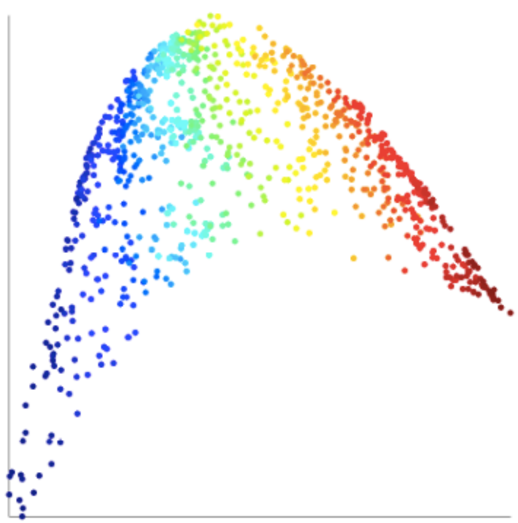
$M = 100$ pixels



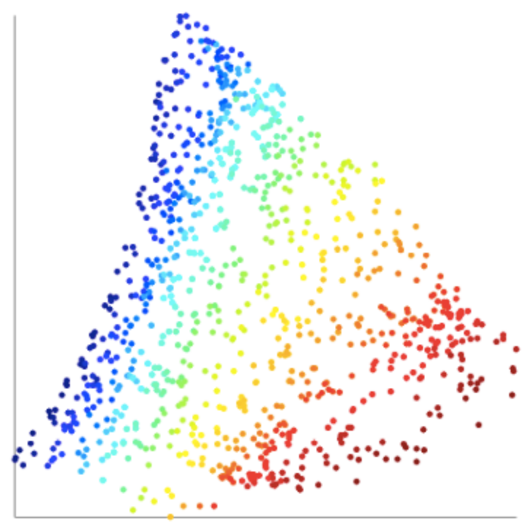
Full Data



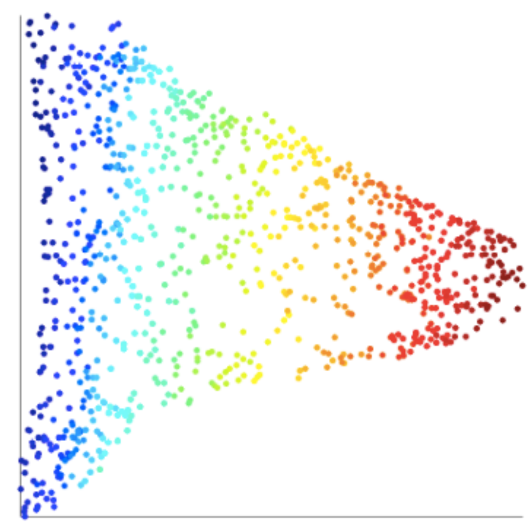
MAPS-LLE



Random

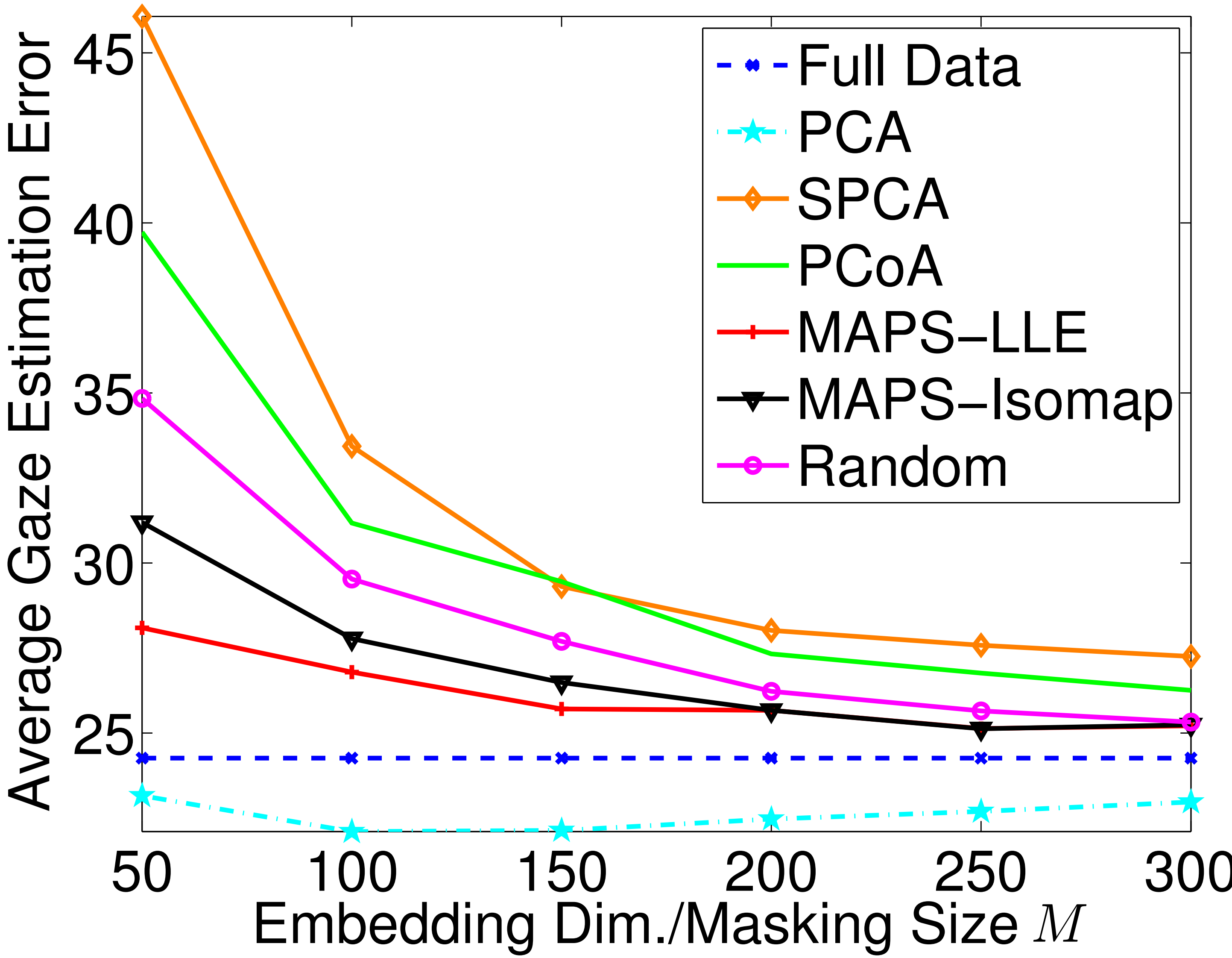


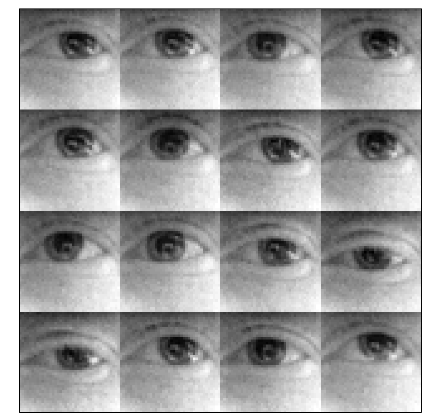
SPCA



PCoA

Computational Eyeglasses: Eye Gaze Tracking





$N = 40 \times 40 = 1600$

Error measured in “target” pixels

Conclusions

- Compressive sensing (CS) for manifold-modeled images via ***random*** or ***customized*** projections (NuMax)
- New sensors enable CS by masking images, i.e., restricting the type of projections
- Our MAPS algorithms find image masks that best ***preserve geometric structure*** used during manifold learning for image datasets
- Greedy algorithms provide good preservation of learned manifold embeddings, suitable for parameter estimation
- While Isomap relies on distances between neighbors, LLE also leverages local geometric structure; different algorithms are optimal for these cases
- Concept of subsampling as ***feature selection*** - supervised and unsupervised learning?

http://www.ecs.umass.edu/~mduarte mduarte@ecs.umass.edu