

Order the Face Images by Manifold Learning

YU Zhiyuan, CHENG Haoyi, QUAN Xueyang

Hong Kong University of Science and Technology

Table of Contents

- ➊ Introduction
- ➋ Dataset
- ➌ Methodology
- ➍ Experiment Results and Analysis
- ➎ Conclusion
- ➏ Reference

Table of Contents

1 Introduction

2 Dataset

3 Methodology

4 Experiment Results and Analysis

5 Conclusion

6 Reference

Introduction

- Ordering orientation of unstructured images
 - ▶ Helps estimate the right motion of object or camera
 - ▶ Applications:
object tracking, multi-view reconstruction, etc.
- Similarity Measurement and Dimensionality Reduction

Objective

- Order the 33 face images using different manifold learning methods
- Compare performances of these methods
- Conduct parameter test for some methods

Table of Contents

- 1 Introduction
- 2 Dataset
- 3 Methodology
- 4 Experiment Results and Analysis
- 5 Conclusion
- 6 Reference



Figure 1: The ground truth ordered face images

Table 1: Labelled rank number of ground truth face images

Image	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Rank	9	13	19	32	6	18	28	7	17	1	5	16	12	10	4	21	22
Image	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	
Rank	26	33	11	2	24	3	27	29	23	14	30	31	20	15	25	8	

The dataset contains 33 face images ($X \in \mathbb{R}^{112 \times 92 \times 33}$).

Table of Contents

- 1 Introduction
- 2 Dataset
- 3 Methodology**
- 4 Experiment Results and Analysis
- 5 Conclusion
- 6 Reference

Methodology

- 6 manifold learning methods
 - ▶ Diffusion Map (DM)
 - ▶ Multi-Dimensional Scaling (MDS)
 - ▶ Isometric Mapping (ISOMAP)
 - ▶ Locally Linear Embedding (LLE)
 - ▶ Local Tangent Space Alignment (LTSA)
 - ▶ t-distributed Stochastic Neighbor Embedding (t-SNE)
- Compare methods
 - ▶ Convert the ground truth label and 6 predicted rank labels into hot matrices.
 - ▶ Calculate the Frobenius norm of the distance between predicted and ground truth hot matrices.
- Test parameters
 - ▶ Number of neighbors to consider for each point: 2, 4, 5, 8, 16, 32.
 - ▶ Test 3 methods: ISOMAP, LLE, and LTSA.

6 Methods I

- DM

- ▶ Dataset, $X \in \mathbb{R}^{n \times p}$ ($\mathbb{R}^{33 \times 10304}$)

Gaussian kernel, $k(x, y) = \exp\left(-\frac{\|x-y\|^2}{\alpha}\right)$

Diffusion matrix, $P = D^{-1}K$

- ▶ Approximate diffusion distance

- MDS

- ▶ Dissimilarity matrix of X , $D_{ij} = \|x_i - x_j\|_2$

- ▶ Optimize

$$\operatorname{argmin}_{p_1, \dots, p_n} \sum_{i < j} (\|p_i - p_j\|_2 - D_{ij})^2$$

6 Methods II

- ISOMAP

- ▶ Construct a neighboring graph G
- ▶ Compute the shortest paths distance $d_G(i, j)$
- ▶ Construct a lower dimensional embedding.
Apply the classical MDS to $D_G = \{d_G(i, j)\}$.

- LLE

- ▶ Local Fit

$$\min \left\| x_i - \sum_{j \in N_i} w_{ij} x_j \right\|^2 \quad \text{s.t.} \quad \sum_{j \in N_i} w_{ij} = 1$$

- ▶ Global Alignment

$$\min_Y \sum_i \left\| y_i - \sum_{j=1}^n W_{ij} y_j \right\|^2$$

6 Methods III

- LTSA
 - ▶ Given $x_1, x_2, \dots, x_n \in \mathbb{R}^p$,
Compute the k-nearest neighbours,
Compute local SVD on neighbourhood
 - ▶ Search the nearest neighbors
 - ▶ Align the tangent space
 - ▶ Do eigenvalue decomposition
- t-SNE
 - ▶ Compute pairwise affinities with perplexity
 - ▶ Sample initial solution
 - ▶ Compute low-dimensional affinities and gradient

Table of Contents

- 1 Introduction
- 2 Dataset
- 3 Methodology
- 4 Experiment Results and Analysis
- 5 Conclusion
- 6 Reference

Table 2: The rank results of various manifold learning methods

Rank	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Ground Truth	10	21	23	15	11	5	8	33	1	14	20	13	2	27	31	12	9
DM	10	21	23	15	11	5	8	1	33	14	20	13	2	27	31	12	9
MDS	10	21	23	19	14	20	5	11	33	8	4	13	29	1	28	15	25
ISOMAP	10	21	23	15	5	11	1	33	14	8	20	13	2	27	31	12	9
LLE	10	21	23	15	5	11	8	1	33	14	20	13	2	27	31	12	9
LTSA	10	21	23	15	11	5	33	8	1	14	20	13	2	31	27	12	9
t-SNE	21	10	3	31	26	23	15	30	6	27	22	13	17	9	1	14	16

Rank	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
Ground Truth	6	3	30	16	17	26	22	32	18	24	7	25	28	29	4	19
DM	6	3	30	16	17	22	26	32	18	24	7	25	29	28	4	19
MDS	27	2	7	24	9	18	31	12	32	6	17	26	30	16	3	22
ISOMAP	6	3	30	16	17	26	22	32	18	24	7	25	28	29	4	19
LLE	6	3	30	16	17	22	26	32	18	24	7	25	28	29	4	19
LTSA	6	3	30	16	17	26	22	32	18	24	7	25	25	29	4	19
t-SNE	18	32	5	12	33	28	2	25	7	8	11	29	24	20	19	4

Table 3: One-hot error of different algorithms

Algorithm	Diffusion Map	MDS	ISOMAP	LLE	LTSA	t-SNE
Error	3.46410	7.74560	3.16228	3.46410	2.82843	7.87401

Results - DM & MDS



Figure 2: DM Scatter

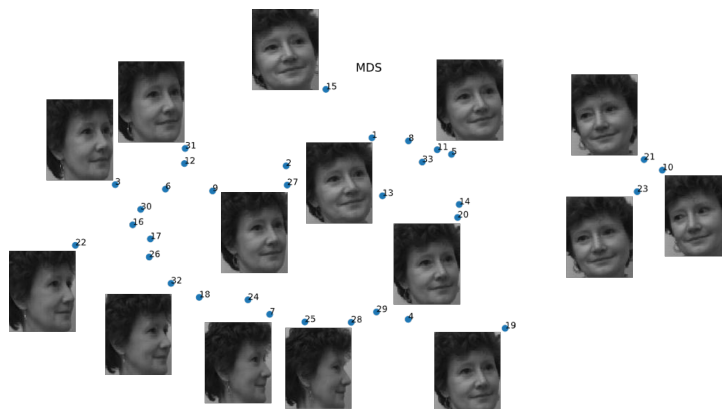


Figure 3: MDS Scatter

Results - ISOMAP & LLE

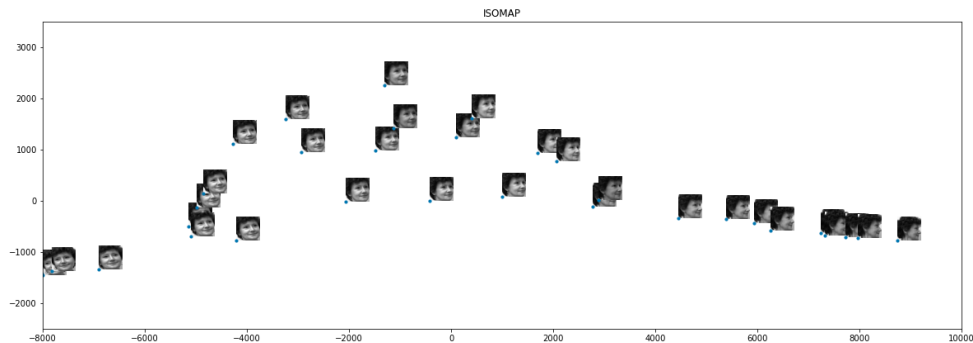


Figure 4: ISOMAP (K=5)

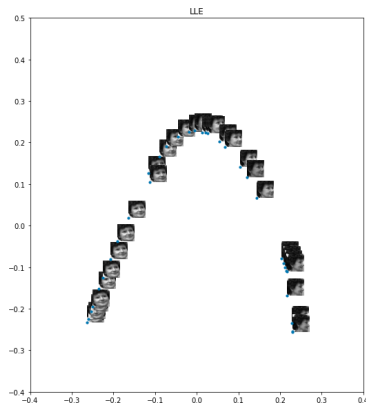


Figure 5: LLE (K=5)

Results - LTSA & t-SNE

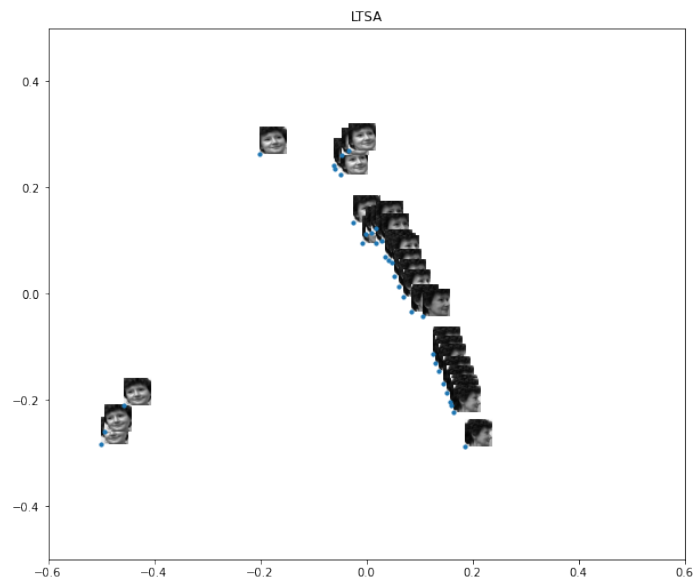


Figure 6: LTSA (K=5)

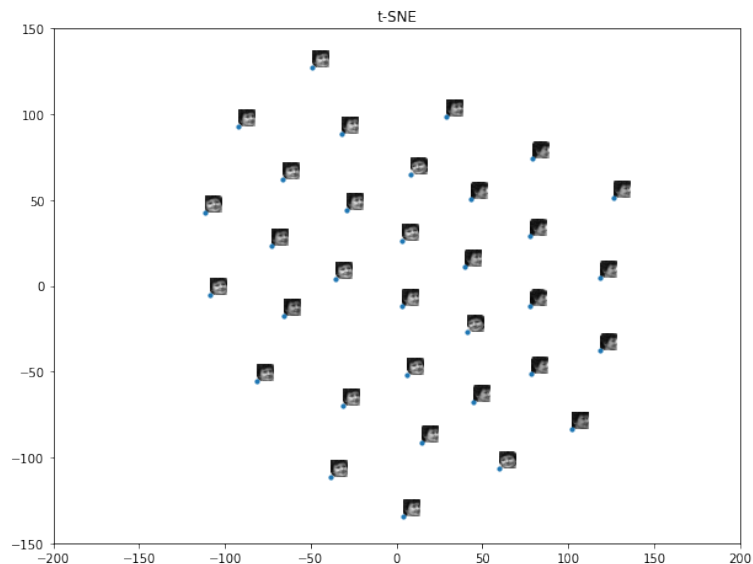


Figure 7: t-SNE

Results - Test Parameters

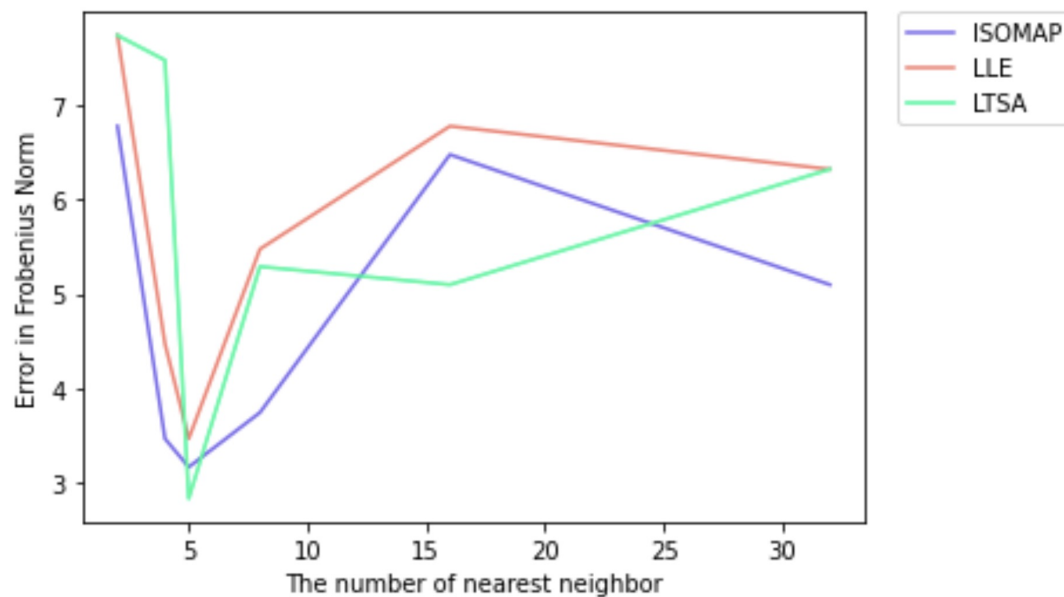


Figure 8: Parameter Test for ISOMAP, LLE, LTSA

Table of Contents

- 1 Introduction
- 2 Dataset
- 3 Methodology
- 4 Experiment Results and Analysis
- 5 Conclusion**
- 6 Reference

Conclusion

- Except the t-SNE and MDS methods, other four nonlinear embedding methods obtain the reasonable images order.
- The more detailed the first eigenvector obtain, the better performance the methods show.
- When the number of nearest neighbour closes to 5, the algorithm perform best, which is true for almost all methods.

Table of Contents

- 1 Introduction
- 2 Dataset
- 3 Methodology
- 4 Experiment Results and Analysis
- 5 Conclusion
- 6 Reference**

Reference

- [1] Joshua B. Tenenbaum, Vin de Silva and John C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. In *Science 22 Dec 2000: Vol. 290, Issue 5500*, pages 2319–2323.
- [2] Sam T. Roweis and Lawrence K. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. In *Science 22 Dec 2000: Vol. 290, Issue 5500*, pages 2323–2326.
- [3] Manifold learning. Retrieved from:
<https://scikit-learn.org/stable/modules/manifold.html>

Contribution

- YU Zhiyuan
 - ▶ Code in Python for algorithms: DM and MDS, order correct ground truth face images
 - ▶ Write report: 1, 2, 3.1 and 3.2
 - ▶ Presentation
- CHENG Haoyi
 - ▶ Code in Python for algorithms: LTSA and t-SNE
 - ▶ Write report: 3.5, 3.6, 4 and 5
 - ▶ Presentation
- QUAN Xueyang
 - ▶ Code in Python for algorithms: ISOMAP and LLE
 - ▶ Write report: 3.3 and 3.4
 - ▶ Write PPT
 - ▶ Presentation