TEAM REPORT

计算机图形学小组展示

流形学习的研究

小组成员

李鹏飞 、陈欢、胡涵 、林葳洁、王凯笛

小组分工

- 使用论文中的算法，测试原论文中例子，验证其可行性，同时自行编写代码，尝试将此方法应用在ORL人脸库的处理上，调整出合适的参数，并将结果显示出来（李鹏飞负责）
- 流形学习 （陈欢负责）
- 阅读LLE (Locally Linear Embedding)和Isomap两篇文章，重点学习并归纳LLE方法。对比LLE和Isomap，总结它们的优点和缺点。（林葳洁负责）
- 例子解析 （胡涵负责）
- 背景介绍及PPT整合（王凯笛负责）

第一阶段	第二阶段	第三阶段
• **阅读文献**	• **分工写代码**	• **整合展示**

Part One
背景介绍

1.降维的必要性和重要性

- 高维数据在各种应用问题中广泛存在。会带来“维数灾难” “空空间现象。
- 降维的目的就是要在保持数据信息损失最少的情况下，发现其内在低维结构，恢复其内在的低维变量，为后续的数据处理提供简洁有效的数据表示。
- 线性降维方法主要是通过寻找原始数据的一个线性映射，将高维数据映射到一个低维的子空间，从而实现降维。（PCA, MDS）

2.主成分分析法PCA

- 基本思想：将原来具有一定相关性的指标进行重新组合，形成一组新的彼此之间互不相关的指标，这些新指标是原来P个指标的线性组合（或称综合性指标），用这组新的指标来代替原来的指标进行后续分析。
- 几何的观点来看，主成分分析是对原坐标轴作一个坐标旋转，得到相互正交的坐标轴,使得该坐标轴的方向为所有数据点分散最开的方向，依据得到的特征值的大小排列这些新的坐标轴。

PCA优缺点

- 优点：良好的可解释性、计算量小。
- 不足
  - 由于PCA对原有数据采用线性变换得到低维坐标，因此当原有数据具有内在扭曲的非线性结构时，得到的降维结果不能反映数据本质结构
  - 虽然能找到方差最大方向，但这未必对分类和识别问题最有利，另一方面其忽略了类标号信息，同样不利于分类和识别
  - PCA对于投影维数难以估计。虽然某些情况下我们可以根据协方差矩阵相邻特征值之间的比值来确定降维维数，但是当特征值变化不明显时，就难以对主成分进行取舍

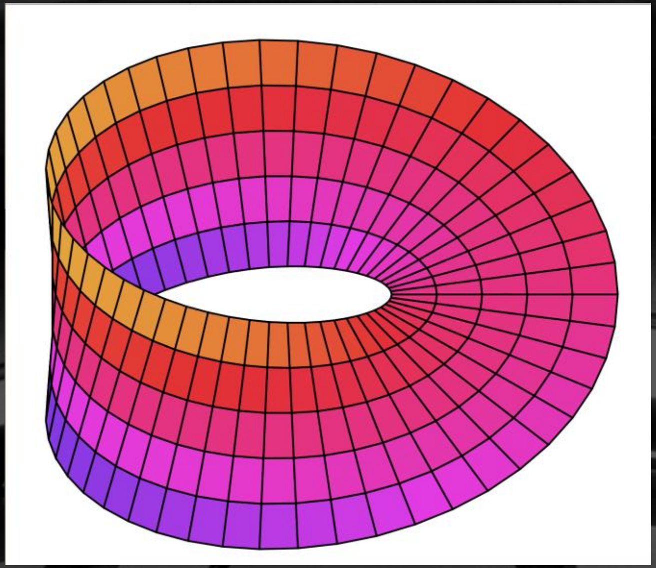
3.多维尺度分析MDS

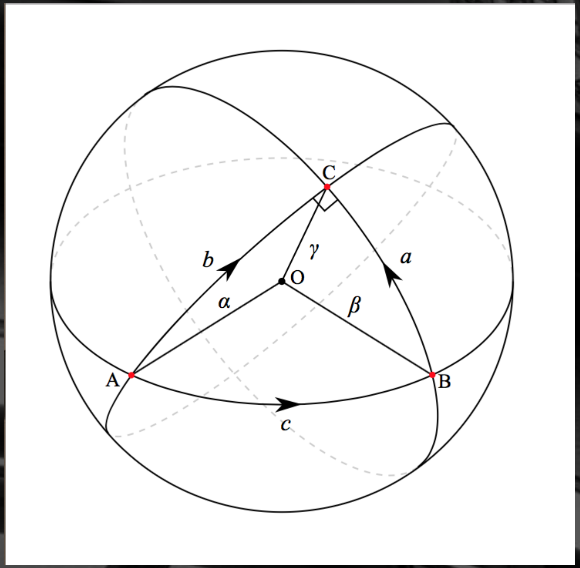
- 对于给定的数据间的不相似性矩阵（距离矩阵），其目标是寻找该数据在低维空间的一种拟合构图，使得数据在低维空间保持该不相似性（距离）。如果给定的距离矩阵为欧氏距离矩阵，则MDS将获得和PCA相同的结果。

Part Two
流形学习

流形与测地线

- 流形：
  - 可以看做是很多平面的叠加，即为局部具有欧式空间的空间。
- 测地线：
  - 弯曲的“直线”，用来计算流形上两点的最短路径





流形学习的基础想法

- 以MDS为基础，但是尝试着保持数据之间存在的内在几何性质，即为点对之间的测地流形距离。
- 关键在于估计相距很远的点之间的测地距离。
- 测地线的近似：
  - 对于邻近的点对而言，可以看做是欧式空间，因此普通的欧氏距离可以认为是测地距离的一个很好的近似；
  - 而对于相隔很远的点而言，测地距离可以被认为由于一系列小的临近点之间的叠加来近似。这里我们常用的方法是图模型之中常用的最短路径算法(Dijkstra, Floyd)。

Isometric Feature Mapping

- Step1: 构建邻接图
  - 对于所有的数据点构建邻接图，若两个点之间的距离小于e，则认为点i,j是相互连接在一起的，或者计算距某一个点最小的K个点。将边的长度定义为d(i,j)
- Step2: 计算最短路径
  - 初始化$d_G(i,j)=d_X(i,j)$, 当i,j由一条边连接时; $d_G(i,j)=\infty$,当i,j两点之间没有连接时。接下来，对于每一个k=1,2,3……N, 将$d_G(i,j)$由$\min(d_G(i,j), d_G(i,k)+d_G(k,j))$。此时，经过更新后的邻接矩阵包含有每两个点之间的最短距离。
- Step3: 构建D维嵌入
  - 令λ_p为$\tau(D_G)$的第p个特征值（τ将两点之间的距离变为内积），并且v_p^i为第p个特征向量的第i个元素。接下来令第i个样本在第p个坐标上的位置为$\sqrt{\lambda_p v_p^i}$

Isometric Mapping的收敛性质

- 正如PCA与MDS所保证的，在足够充分的数据的情况下，可以恢复原先的线性流形结构; Isometric可以在有充分数据的情况下，恢复一系列的非线性流形。
- Isometric Mapping的近似性质依赖于当数据点的数量增加的时候，$d_G(i,j)$给出了$d_M(i,j)$的一个测地线距离的近似。
- **近似收敛定理：**给定$\lambda_1,\lambda_2,\mu>0$, 那么对于某一个概率分布α，当样本量足够大的时候

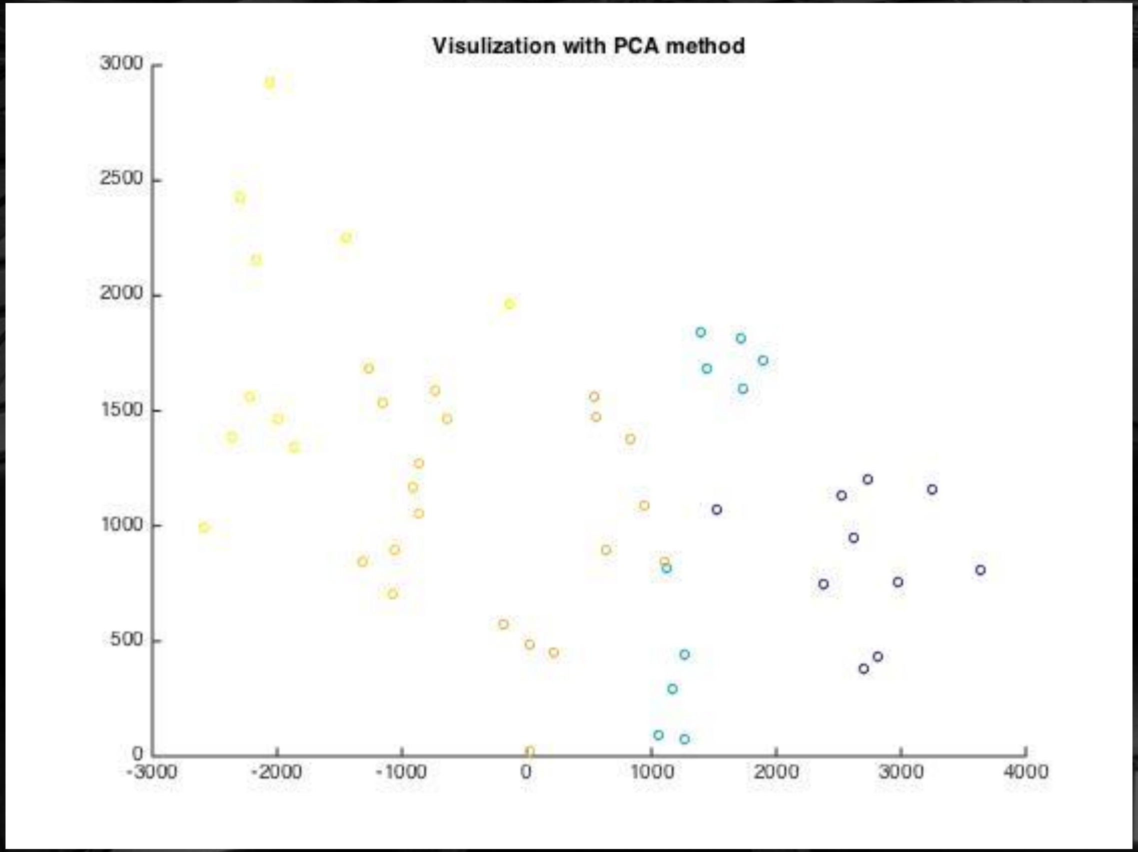
$$1-\lambda_1 \leq \frac{graph\ distance}{geodesic\ distance} \leq 1+\lambda_2$$

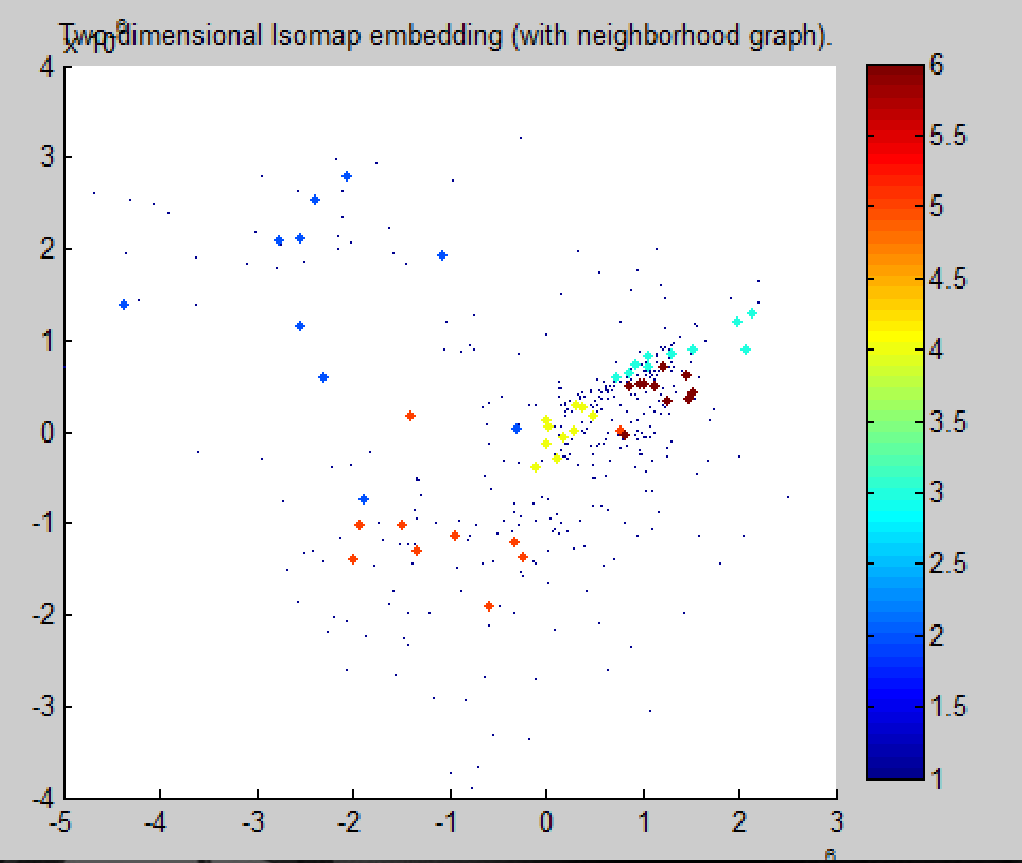
以一个至少为1-μ的概率成立

Isometric Mapping的优势

- 先前方法的缺点:
  - 局部线性化方法难以表示一个数据集所具有的整体结构;
  - 非线性技术基于贪婪优化的方法，虽然可以捕捉到全局的结构，但是没有足够好的算法来进行计算;
- Isometric Mapping的优点
  - Isometric Mapping可以采用非迭代，多项式时间保证全局的最优；
  - 对于欧式内蕴流形而言，可以保证其近似收敛到真正的结构；
  - 可以发现流形自身所具有的结构，而不需要在计算之前决定采用流形的维度。

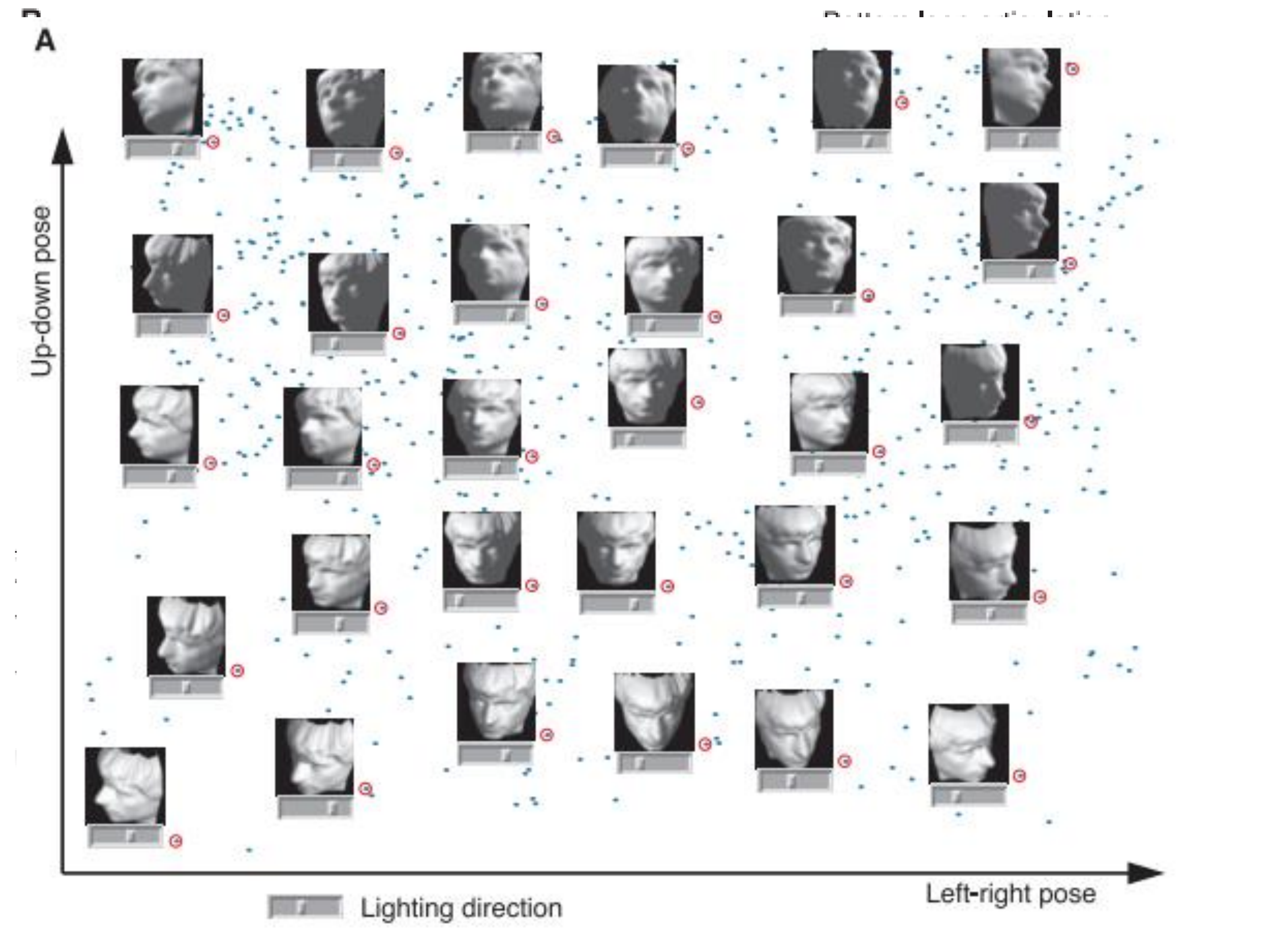
PCA与Isometric可视化的结果对比

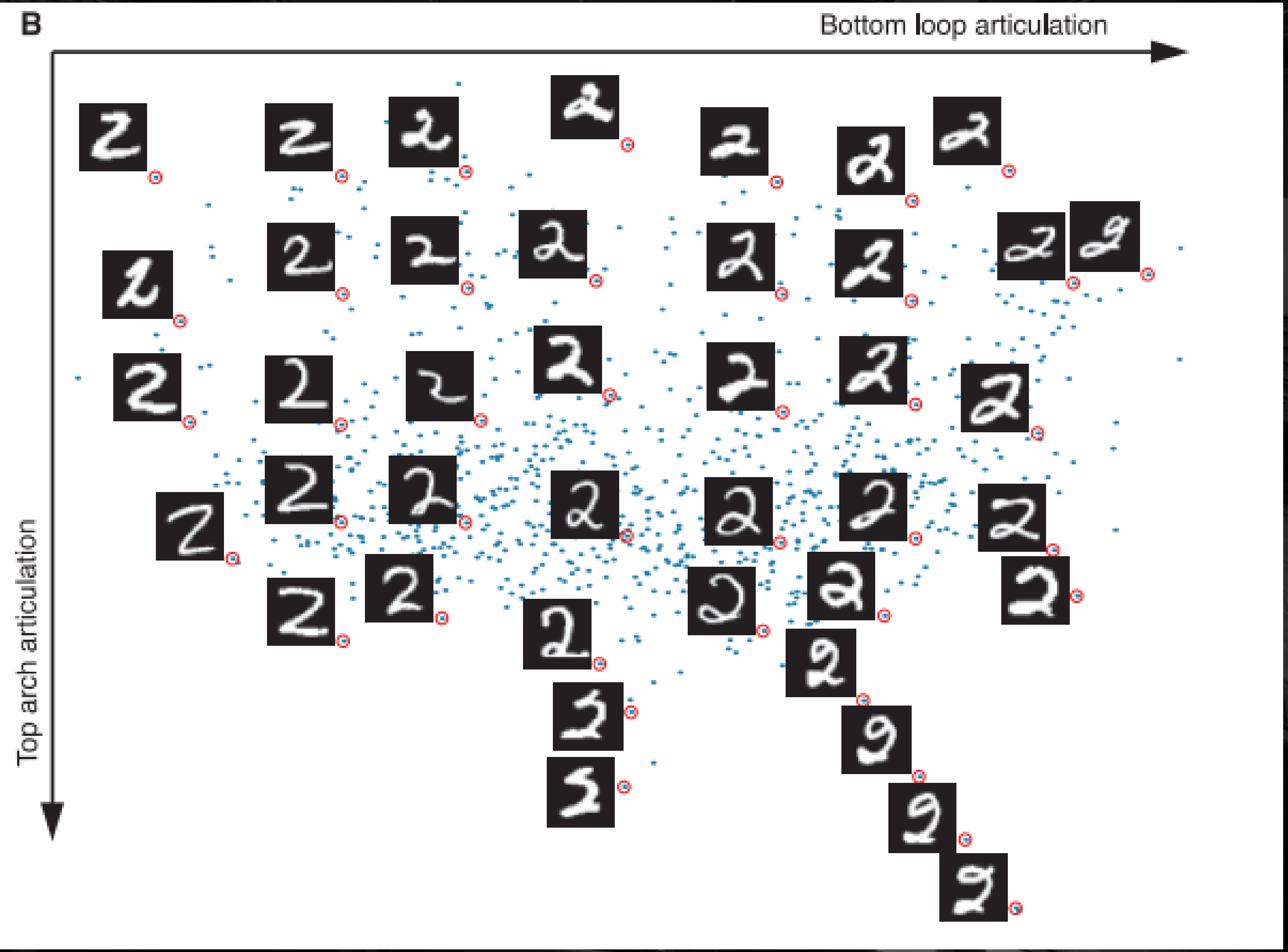




Part Three
例子解析

Step		
1	Construct neighborhood graph	Define the graph G over all data points by connecting points i and j if [as measured by $d_X(i,j)$] they are closer than ϵ (ϵ-Isomap), or if i is one of the K nearest neighbors of j (K-Isomap). Set edge lengths equal to $d_X(i,j)$.
2	Compute shortest paths	Initialize $d_G(i,j) = d_X(i,j)$ if i,j are linked by an edge; $d_G(i,j) = \infty$ otherwise. Then for each value of $k = 1, 2, \ldots, N$ in turn, replace all entries $d_G(i,j)$ by $\min\{d_G(i,j), d_G(i,k) + d_G(k,j)\}$. The matrix of final values $D_G = \{d_G(i,j)\}$ will contain the shortest path distances between all pairs of points in G (*16, 19*).
3	Construct d-dimensional embedding	Let λ_p be the p-th eigenvalue (in decreasing order) of the matrix $\tau(D_G)$ (*17*), and v_p^i be the i-th component of the p-th eigenvector. Then set the p-th component of the d-dimensional coordinate vector $\mathbf{y}_i$ equal to $\sqrt{\lambda_p} v_p^i$.

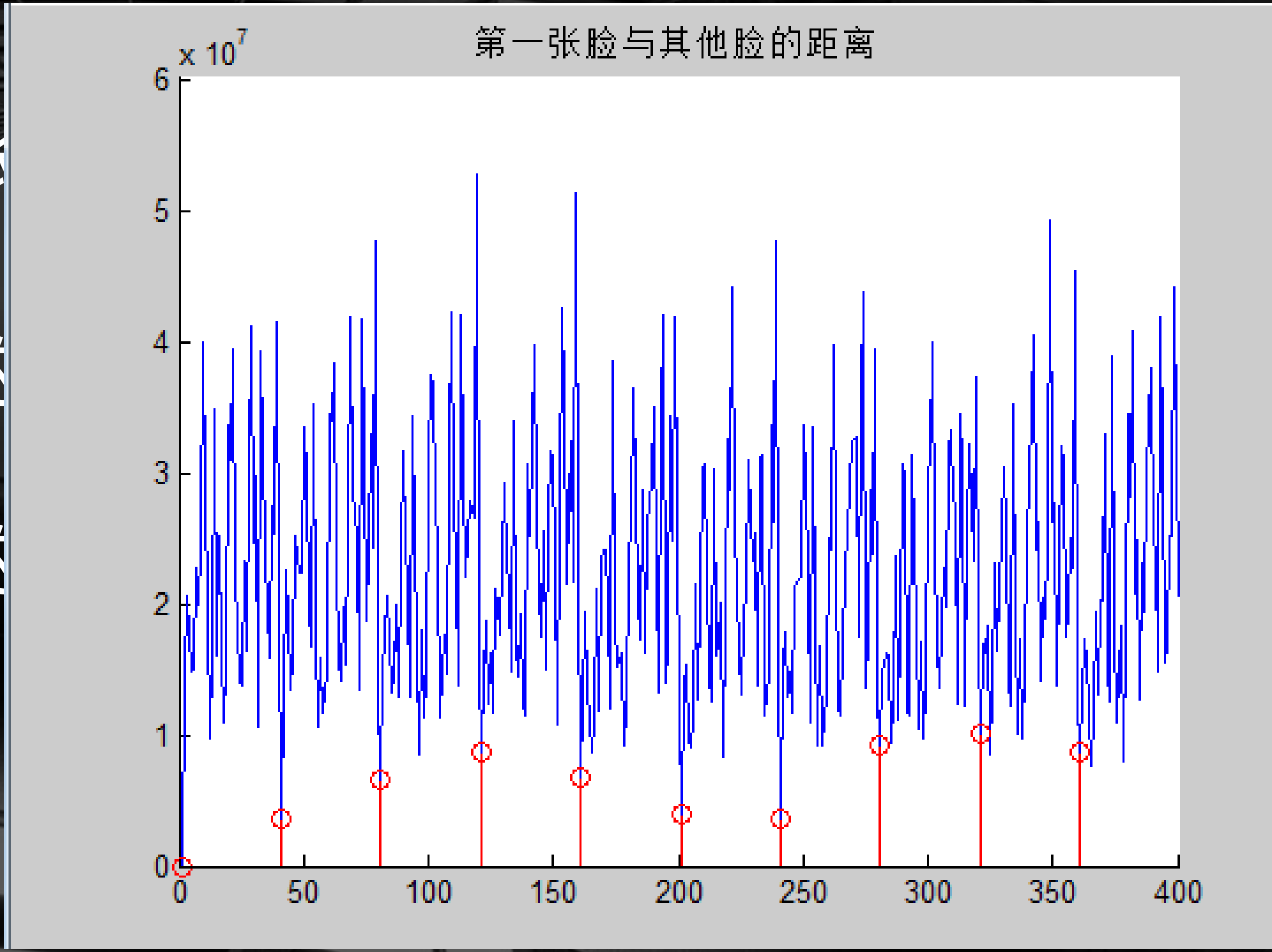

A
Up-down pose
Left-right pose
Lighting direction


B
Bottom loop articulation
Top arch articulation

Part Four
数据测试

导入之前人脸识别的数据库

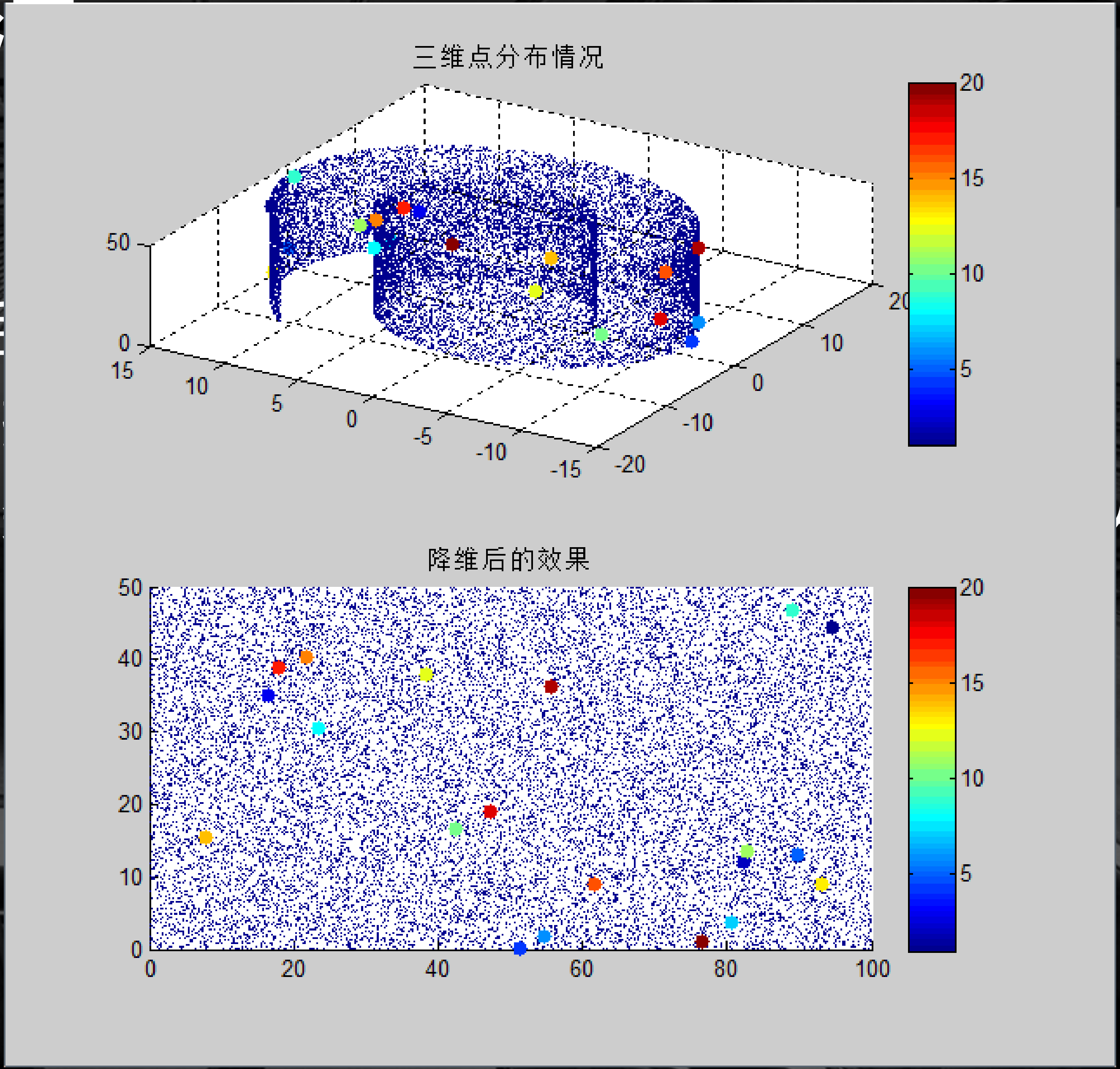
- 将所有的人脸图像读出来，
- 将每个矩阵化为一行，每张
- 求取这些向量的欧氏距离
- 由于距离矩阵必然是对称的
  量
- 同一个人的脸的图像之间的



- 具体计算数据见 人脸数据.mat 文件

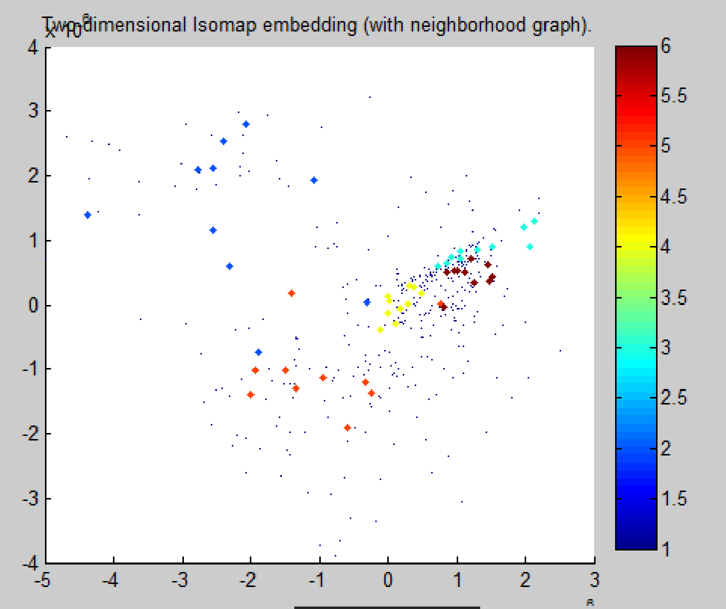
使用isomap方法

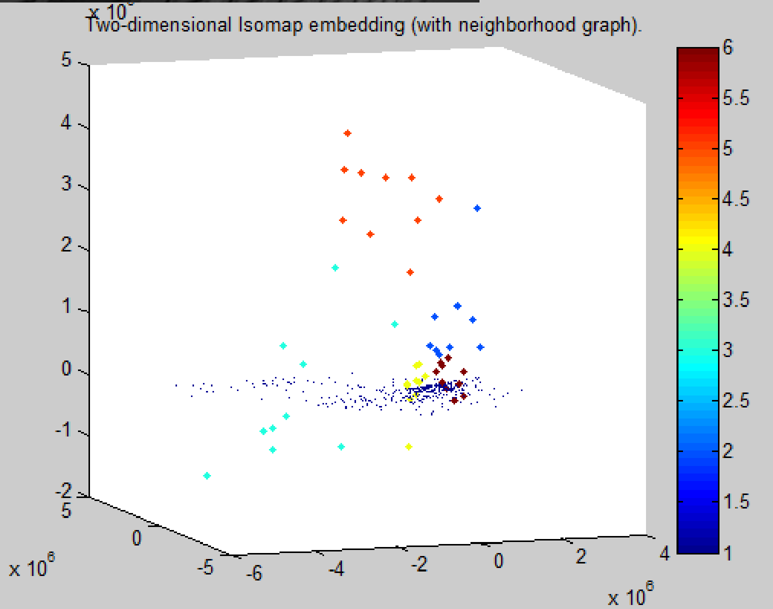
- 首先测试论文中所使用
- 将结果计算出来 ，然后
- 为使效果更清晰，我们
- 效果非常理想，在这种 … 从的

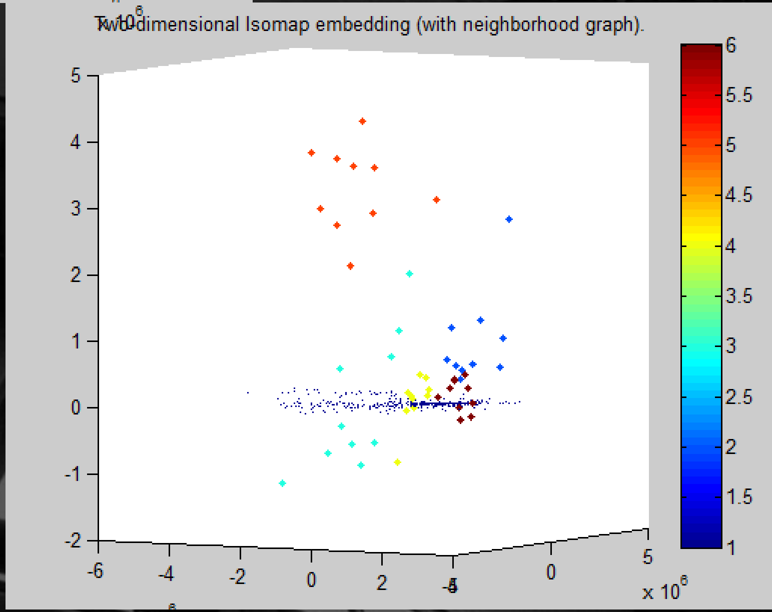


- 具体代码见附件

测试新数据


Two-dimensional Isomap embedding (with neighborhood graph).

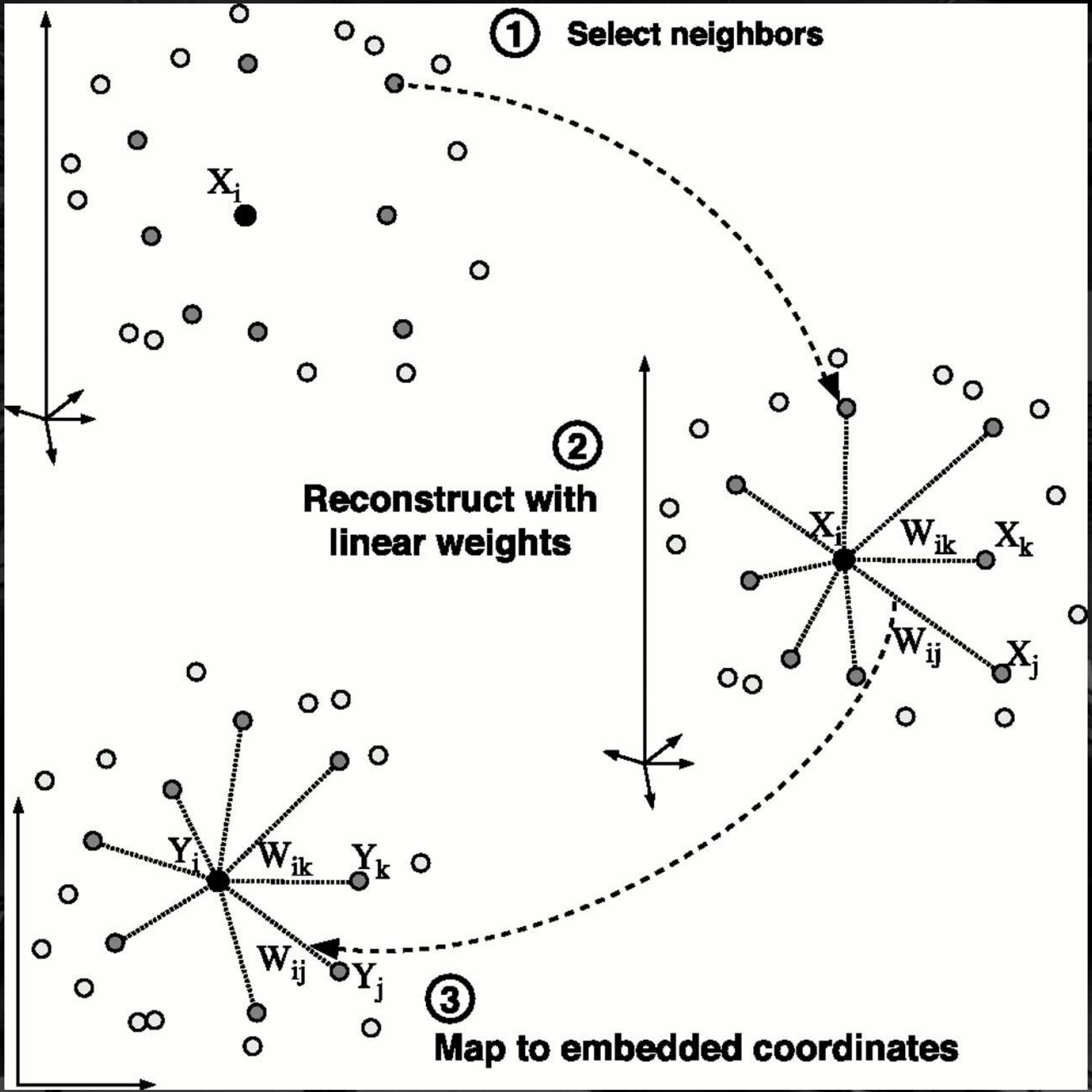

Two-dimensional Isomap embedding (with neighborhood graph).


Two-dimensional Isomap embedding (with neighborhood graph).

Part Five
LLE算法

LLE (Locally linear embedding)

- 前提假设
  - 采样数据所在的低维流形在局部是线性的
  - 每个采样点均可以利用其近邻样本进行线性重构表示
- 学习目标
  - 低维空间中保持每个邻域中的重构权值不变
  - 在嵌入映射为局部线性的条件下，最小化重构误差
  - 最终形式化为特征值分解问题



1 Select neighbors
X_i
2 Reconstruct with linear weights
X_i
W_{ik}
X_k
W_{ij}
X_j
Y_i
W_{ik}
Y_k
W_{ij}
Y_j
3 Map to embedded coordinates

LLE算法流程

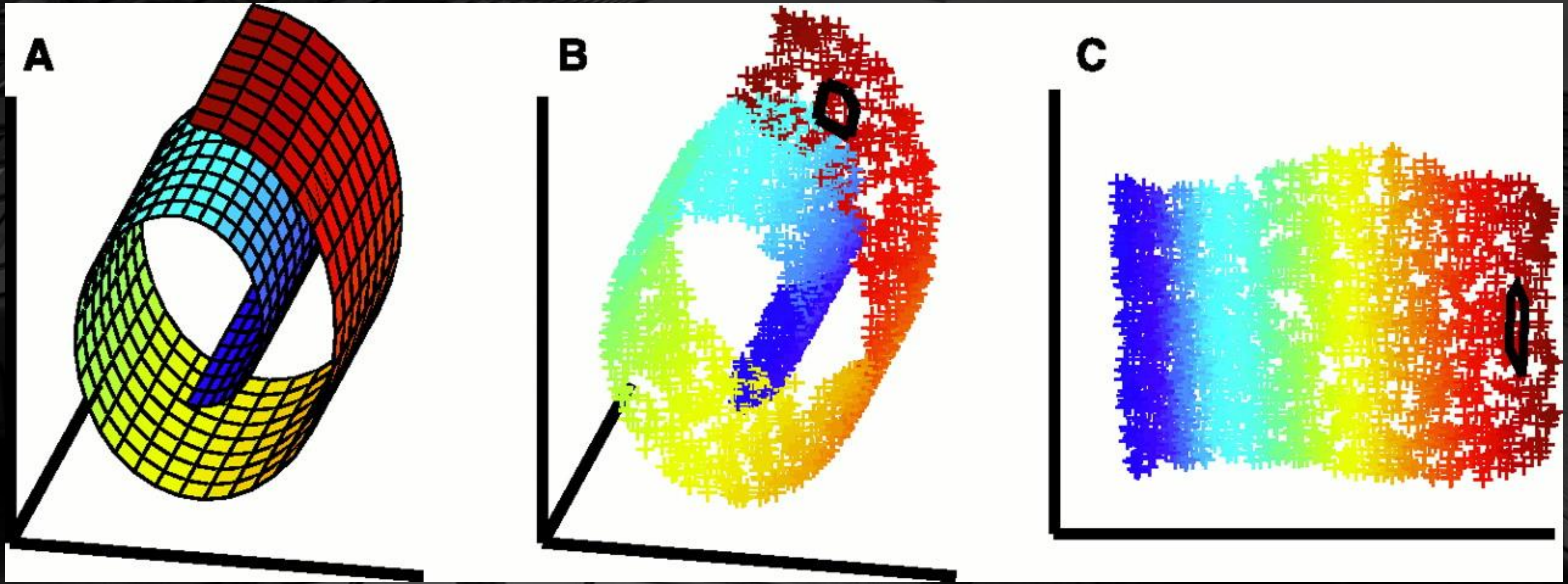
- 1 计算每一个点X_i的近邻点，一般采用K近邻或者ε邻域.
- 2 计算权值W_{ij}使得把X_i用它的K个近邻点线性表示的误差最小，即通过最小化$\|X_i - W_{ij}X_j\|$来求出W_{ij}.

$$\min \varepsilon(W) = \sum_i \left\| x_i - \sum_j w_{ij} x_{ij} \right\|^2$$

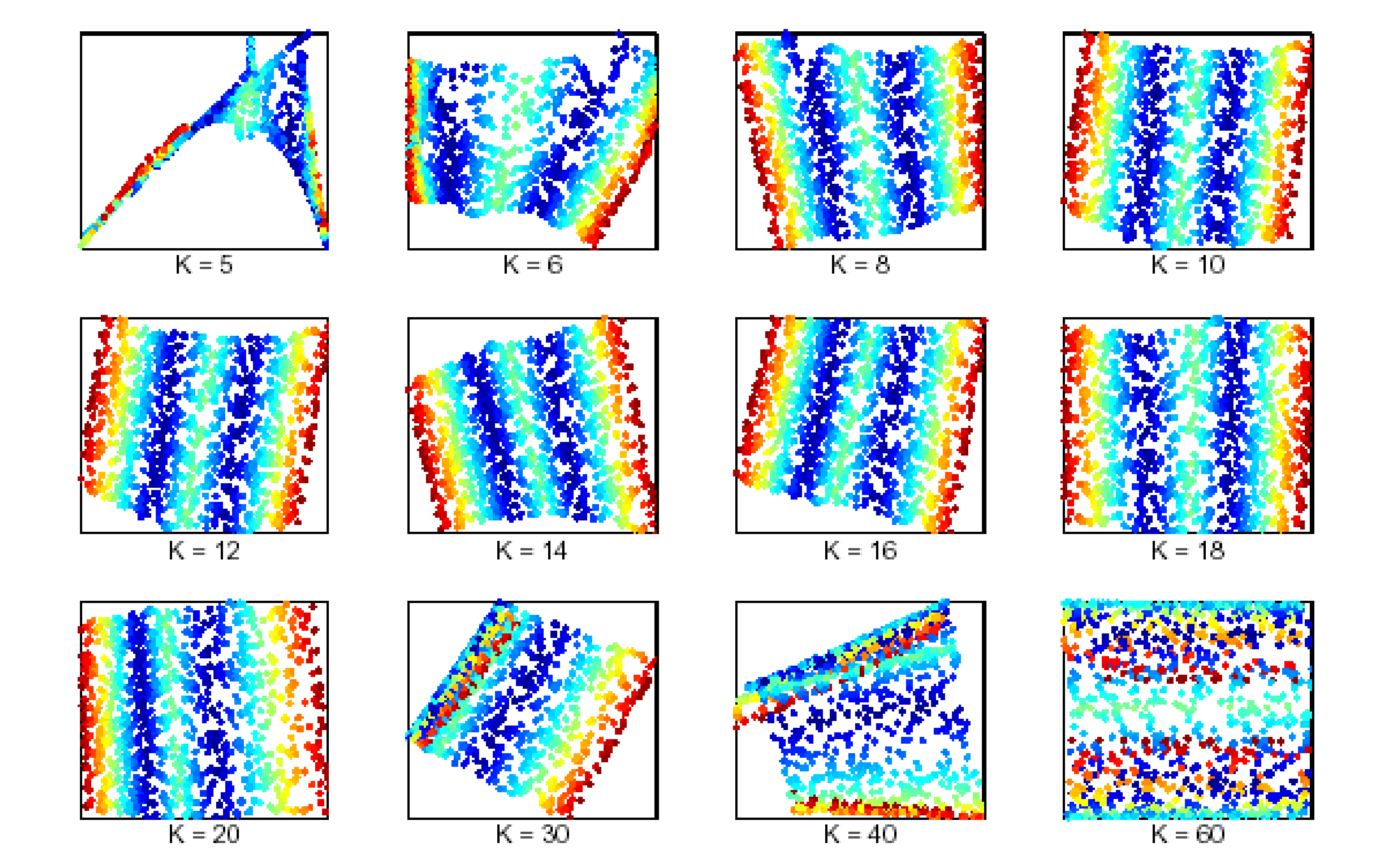
- 3 保持权值W_{ij}不变，求X_i在低维空间的象Y_i，使得低维重构误差最小.

$$\min \varphi(W) = \sum_i \left\| y_i - \sum_j w_{ij} y_{ij} \right\|^2$$

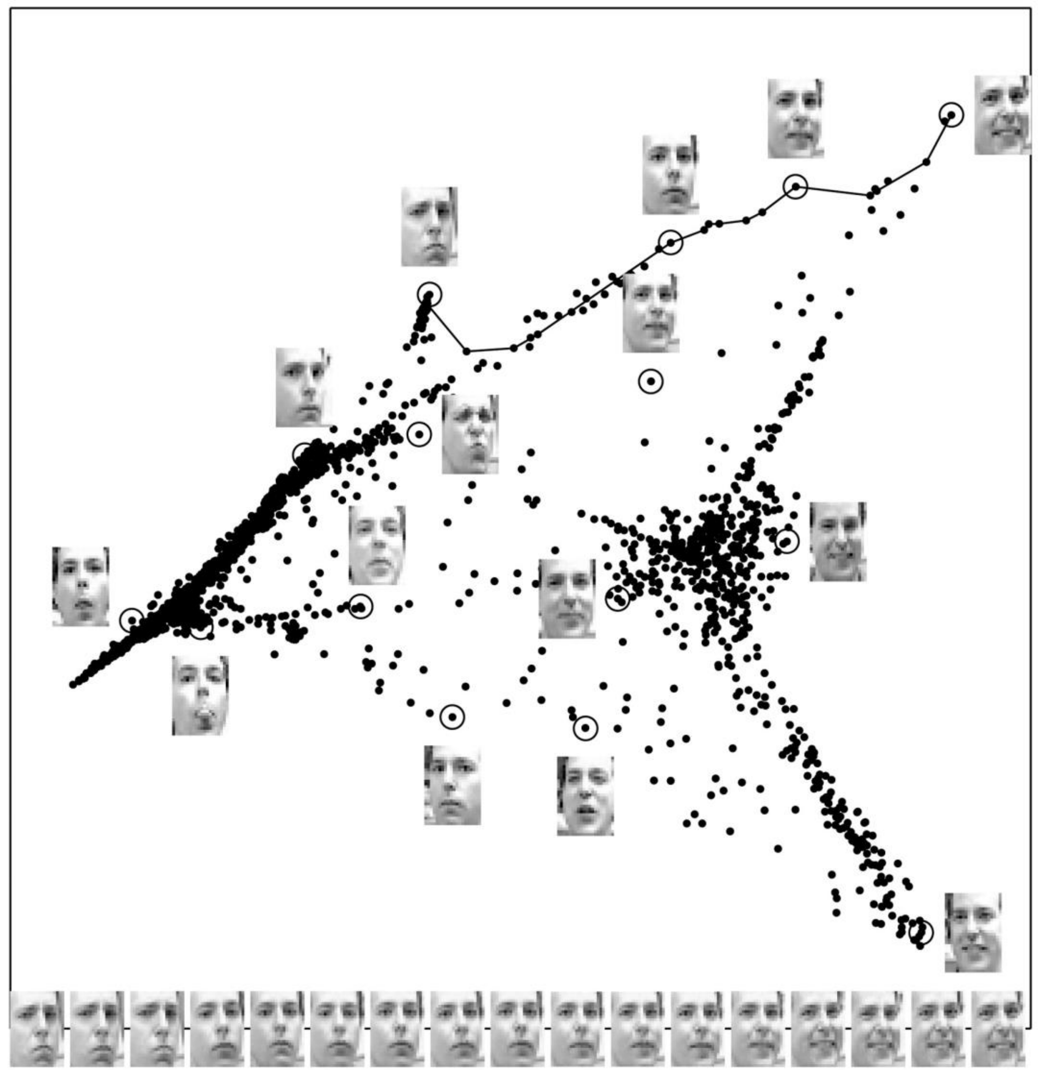
LLE实验结果



LLE实验结果



LLE实验结果



ISOMAP	LLE
利用最短路径逼近测地距离	以局部线性来保持整体的拓扑结构
全局	局部
计算点对间的最短路径比较耗时	稀疏矩阵特征值计算，计算复杂度相对较小
不适于学习有较大内在曲率的流形	对样本中的噪声和邻域参数比较敏感

谢谢！