ENSEMBLES OF LANDMARK MULTIDIMENSIONAL SCALINGS

Seunghak Lee † *and Seungjin Choi* §

† Department of Computer Science, University of Toronto, Canada
seunghak@cs.toronto.edu
§ Department of Computer Science, POSTECH, Korea
seungjin@postech.ac.kr

ABSTRACT

Landmark multidimensional scaling (LMDS) uses a subset of data (landmark points) to solve classical MDS, where the scalability is increased but the approximation is noise-sensitive. In this paper we present an ensemble of LMDSs, referred to as landmark MDS ensemble (LMDSE), where we use a portion of the input in a piecewise manner to solve classical MDS, combining individual LMDS solutions which operate on different partitions of the input. Ground control points (GCPs) that are shared by partitions considered in the ensemble, allow us to align individual LMDS solutions in a common coordinate system through affine transformations. LMDSE solution is determined by averaging aligned LMDS solutions. We show that LMDSE is less noise-sensitive while maintaining the scalability as well as the speed of LMDS. Experiments on synthetic data (noisy grid) and real-world data (similar image retrieval) confirm the high performance of the proposed LMDSE.

Index Terms— Dimensionality reduction, embedding, multidimensional scaling (MDS), unsupervised learning

1. INTRODUCTION

Multidimensional scaling (MDS) is a widely-used method for metric-preserving dimensionality reduction, the goal of which is to embed objects in a low-dimensional Euclidean space, taking pairwise dissimilarities as input while preserving underlying geometry [1]. Classical MDS (classical scaling) is a representative method, where the embedding is performed by the eigendecomposition of a Gram matrix derived from the input distance matrix. The bottleneck in the classical MDS is the eigendecomposition of an $N \times N$ Gram matrix (N is the number of data points), which costs $\mathcal{O}(N^3)$ complexity in time.

Various methods have been developed to address the scalability, including FastMap [3], MetricMap [7], and LMDS [2]. All these three methods belong to a class of Nyström approximation and it was reported that LMDS outperformed the other two methods and was the fastest [6]. Recent divide-and-conquer approach for fast embedding includes [8, 4]. LMDS [2] uses a subset of input, referred to as *landmark points*, the low-dimensional embedding of which enables remaining points to be embedded using their distances to landmark points. LMDS substantially reduces time and space complexity from $\mathcal{O}(CN^2 + N^3)$ and $\mathcal{O}(N^2)$ to $\mathcal{O}(ClN + nlN + l^3)$ and $\mathcal{O}(lN)$ respectively, where N is the number of data points, l is the number of landmark points, n is the dimension of the low-dimensional space, and C is the cost to compute each entry of the distance matrix [2].

LMDS dramatically improves the scalability, but is known to be more sensitive to noise or outliers, compared to classical MDS. In this letter, we present a divide-and-conquer method where we use all of the input but in a piecewise manner. The method is referred to as *LMDSE* where we combine individual LMDS solutions which operate on different partitions of the input, through affine transformations based on ground control points (GCPs) which are shared by every partitions of the input. We show that LMDSE dramatically improves the noise-robustness, through experiments on synthetic data (noisy grid) and real-world data (similar image retrieval).

2. THE PROPOSED METHOD

We begin with a brief overview of LMDS, the details about which are found in [2, 6]. Then we present the main contribution, which is a method of LMDSE.

2.1. Landmark MDS

We are given a set of N data points, $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{m \times N}$, and its associated distance (dissimilarity) matrix $\boldsymbol{D} \in \mathbb{R}^{N \times N}$, the (i, j)-entry of which corresponds to the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. In LMDS, classical MDS is applied to the l landmark points, $\boldsymbol{X}_l \in \mathbb{R}^{m \times l}$ ($\boldsymbol{X} = [\boldsymbol{X}_l, \boldsymbol{X}_*]$ and $\boldsymbol{X}_* \in \mathbb{R}^{m \times (N-l)}$) to find their embedding $\boldsymbol{Y}_l \in \mathbb{R}^{n \times l}$ in a low-dimensional Euclidean space, $\mathbb{R}^n$ ($n \leq m$). The distance matrix $\boldsymbol{D}$ is partitioned as shown in the left panel of Fig. 1, where $\boldsymbol{E} \in \mathbb{R}^{l \times l}$ corresponds to the distance matrix involving landmark points. The Gram matrix $\boldsymbol{K} = -\frac{1}{2}\boldsymbol{H}\boldsymbol{D}^2\boldsymbol{H}$ ($\boldsymbol{D}^2 = [D_{ij}^2]$) is also partitioned in the same way, as shown in the right panel of Fig. 1, where $\boldsymbol{H}$ is the centering matrix, $H_{ij} = \delta_{ij} - \frac{1}{N}$. The low-dimensional embedding $\boldsymbol{Y}_l$ is determined by the eigendecomposition (rank-n approximation) of $\boldsymbol{A}$,

$$\boldsymbol{A} \approx \boldsymbol{U}_l \boldsymbol{\Lambda}_l \boldsymbol{U}_l^\top = \boldsymbol{Y}_l^\top \boldsymbol{Y}_l, \tag{1}$$

where $\boldsymbol{U}_l \in \mathbb{R}^{l \times n}$ and $\boldsymbol{\Lambda}_l \in \mathbb{R}^{n \times n}$ are the eigenvector and eigenvalue matrices, leading to

$$\boldsymbol{Y}_l = \boldsymbol{\Lambda}_l^{\frac{1}{2}} \boldsymbol{U}_l^\top. \tag{2}$$

The low-dimensional embedding $\boldsymbol{Y}_* \in \mathbb{R}^{n \times (N-l)}$ of the remaining data points $\boldsymbol{X}_* \in \mathbb{R}^{m \times (N-l)}$ are computed using the distance-based triangulation (which turns out to be similar to Nyström approximation), i.e.,

$$\boldsymbol{Y}_* = \boldsymbol{Y}_l^\dagger \boldsymbol{B} = \boldsymbol{\Lambda}_l^{-\frac{1}{2}} \boldsymbol{U}_l^\top \boldsymbol{B}, \tag{3}$$

where $\boldsymbol{Y}_l^\dagger$ is the pseudo-inverse of $\boldsymbol{Y}_l$ and (i, j)-entry of $\boldsymbol{B}$ is computed using only $\boldsymbol{E}$ and $\boldsymbol{F}$:

$$B_{ij} = -\frac{1}{2}\left\{ F_{ij}^2 - \frac{1}{l}\sum_{j=1}^{l} E_{ij}^2 \right\}.$$

$$D = \begin{array}{|c|c|} \hline \boldsymbol{E} & \boldsymbol{F} \\ \hline \boldsymbol{F}^\top & \boldsymbol{G} \\ \hline \end{array} \qquad K = \begin{array}{|c|c|} \hline \boldsymbol{A} & \boldsymbol{B} \\ \hline \boldsymbol{B}^\top & \boldsymbol{C} \\ \hline \end{array}$$

Fig. 1. Without loss of generality, permute distances between landmark points (denoted by $\boldsymbol{E}$) to be the first rows and columns of the distance matrix $\boldsymbol{D}$. Left panel shows the partitioning of $\boldsymbol{D}$. Right panel shows the corresponding partitioning of the Gram matrix $\boldsymbol{K}$.

2.2. Landmark MDS Ensemble

Now we illustrate our main contribution, LMDSE. We first consider L partitions of the distance matrix $\boldsymbol{D}$, where L is at most $\lfloor N/l \rfloor$, as shown in Fig. 2 in the case of $L = 4$. Each $\boldsymbol{D}_i \in \mathbb{R}^{l \times N}$ corresponds to $[\boldsymbol{E}, \boldsymbol{F}]$ in Fig. 1, while different sets of landmark points contribute $\boldsymbol{D}_i$. We use $\boldsymbol{D}_i$ to find the MDS solution $\boldsymbol{Y}_i = [\boldsymbol{Y}_{i,l}, \boldsymbol{Y}_{i,*}] \in \mathbb{R}^{n \times N}$, where $\boldsymbol{Y}_{i,l} \in \mathbb{R}^{n \times l}$ is the directly-computed low-dimensional embedding and $\boldsymbol{Y}_{i,*} \in \mathbb{R}^{n \times (N-l)}$ is the embedding computed by the distance-based triangulation as in Eq. (3).

$$D = \begin{array}{|c|c|c|c|} \hline \boldsymbol{D}_1 & & & \\ \hline & \boldsymbol{D}_2 & & \\ \hline & & \boldsymbol{D}_3 & \\ \hline & & & \boldsymbol{D}_4 \\ \hline \end{array}$$

Fig. 2. The partitioning of the distance matrix $\boldsymbol{D}$ is shown, where $\boldsymbol{D}_i$ contains distances between l landmark points in a subset of the input and all of N points in the input.

We align L individual LMDS solutions $\boldsymbol{Y}_i$ in a common coordinate in order to combine them to determine an ensemble solution. To this end, we resort to *ground control points* (GCPs) that are shared by all partitions of the input. We denote by $\boldsymbol{Y}_i^c$ the low-dimensional embedding of GCPs using $\boldsymbol{D}_i$. Define $\boldsymbol{Y}_i^{\setminus c} = \boldsymbol{Y}_i - \boldsymbol{Y}_i^c$. The number of GCPs should be at least $n+1$, however, in practice, we choose a sufficient number of GCPs so that they are well distributed across all the landmark sets. Without loss of generality, we choose $\boldsymbol{Y}_1^c$ as reference. Then we compute an affine transformation $\boldsymbol{\Phi}_i$

$$\boldsymbol{\Phi}_i = \begin{bmatrix} \boldsymbol{A}_i & \boldsymbol{\alpha}_i \\ \boldsymbol{0}^\top & 1 \end{bmatrix}, \tag{4}$$

for each $\boldsymbol{Y}_i^c$, $i = 2, \ldots, L$, to line up $\boldsymbol{Y}_i$'s in a common coordinate, which satisfies

$$\begin{bmatrix} \boldsymbol{A}_i & \boldsymbol{\alpha}_i \\ \boldsymbol{0}^\top & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{Y}_i^c \\ \boldsymbol{1}^\top \end{bmatrix} = \begin{bmatrix} \boldsymbol{Y}_1^c \\ \boldsymbol{1}^\top \end{bmatrix}, \tag{5}$$

where $\boldsymbol{1} = [1, \ldots, 1]^\top$ and $\boldsymbol{0} = [0, \ldots, 0]^\top$ with appropriate dimension (in this case, the dimension is the number of GCPs, denoted by p).

Let us denote by $\widetilde{\boldsymbol{Y}}_i = \left\{ \widetilde{\boldsymbol{Y}}_i^c, \widetilde{\boldsymbol{Y}}_i^{\setminus c} \right\}$ the low-dimensional embedding in the common coordinate, where $\widetilde{\boldsymbol{Y}}_i^{\setminus c} = \boldsymbol{A}_i \boldsymbol{Y}_i^{\setminus c} + \boldsymbol{\alpha}_i \boldsymbol{1}^\top$ and $\widetilde{\boldsymbol{Y}}_i^c = \boldsymbol{A}_i \boldsymbol{Y}_i^c + \boldsymbol{\alpha}_i \boldsymbol{1}^\top$ for $i = 1, 2, \ldots, L$. It should be noted that an affine transformation preserves collinearity and ratio of distances. This property is appealing for LMDSE, since the geometry is not much changed through the transformation.

Finally, the LMDSE solution $\widetilde{\boldsymbol{Y}}$ is obtained by averaging L individual solutions $\widetilde{\boldsymbol{Y}}_i$ determined by the affine transformations,

$$\widetilde{\boldsymbol{Y}} = \frac{1}{L} \sum_{i=1}^{L} \widetilde{\boldsymbol{Y}}_i. \tag{6}$$

LMDSE costs $\mathcal{O}(L(ClN + nlN + l^3) + Np^2)$ (p is the number of GCPs), where $\mathcal{O}(Np^2)$ is the additional complexity required by the affine transformation. Since p is a much smaller number than N, $\mathcal{O}(Np^2)$ is considered as scaling linearly with respect to N. Consequently, the time complexity of LMDSE is just L times that of LMDS. Memory complexity in LMDSE is the same as that in LMDS. LMDSE is summarized in a pictorial form as shown in Fig. 3 and its algorithm is outlined in Table 1.

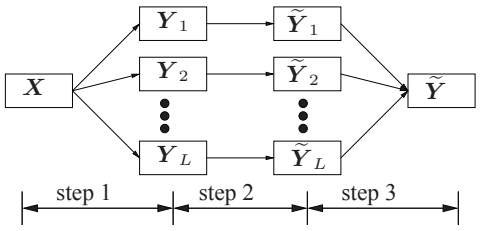

Fig. 3. The schematic diagram for LMDSE is shown, where each step is summarized in Table 1.

Table 1. Algorithm outline: Landmark MDS Ensemble

Input
- $\boldsymbol{X} \in \mathbb{R}^{m \times N}$ or $\boldsymbol{D} \in \mathbb{R}^{N \times N}$
- l: the number of landmark points
- p: the number of GCPs

Output
- $\widetilde{\boldsymbol{Y}} \in \mathbb{R}^{n \times N}$

Step 1
1. Compute L individual LMDS solutions $\boldsymbol{Y}_i$ using $\boldsymbol{D}_i$ for $i = 1, \ldots, L$.

Step 2
2. Randomly select p number of GCPs.
3. Compute affine transformation $\boldsymbol{\Phi}_i$ with choosing $\boldsymbol{Y}_1^c$ as reference, leading to $\widetilde{\boldsymbol{Y}}_i$.

Step 3
4. Determine the solution $\widetilde{\boldsymbol{Y}}$ by averaging $\widetilde{\boldsymbol{Y}}_i$'s.

3. EXPERIMENTS

We present two experiments on noisy grid and similar image retrieval, in order to emphasize the noise-robustness behavior of LMDSE.

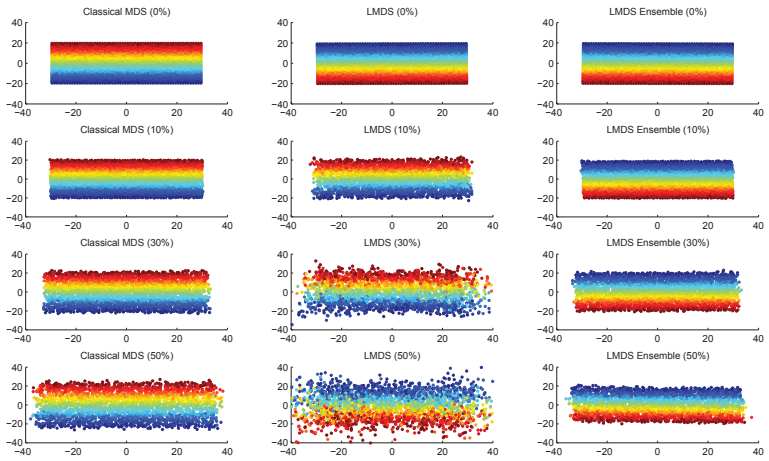

Fig. 4. In the case of noisy grid data, embedding solutions by classical MDS, LMDS (LMDS), and LMDSE are shown. From the top to the bottom, noise levels added to the distance matrix increase: 0%, 10%, 30%, and 50%. Classical MDS and LMDSE faithfully preserve the underlying geometry, whereas LMDS is sensitive to the high level of noise.

3.1. Experiment 1: Noisy Grid

We use the same noisy grid data as used in [2], in order to emphasize the noise-robustness of our LMDSE. The data set contains 2400 data points in a rectangular grid of aspect ratio 1.5. To add noise, we multiply the distance matrix $\boldsymbol{D}$ elementwise by independent random variables drawn from $\exp\left\{\mathcal{N}(0,\sigma^2)\right\}$ where $\mathcal{N}(0,\sigma^2)$ represents Gaussian distribution with mean 0 and variance σ^2. The noise level is defined by by $\nu\% = 100\left(\exp\{\sigma^2\} - 1\right)$ (see Fig. 4). After adding noise, we did not symmetrize the distance matrix to fully keep the perturbation induced by the noise. In this way, more perturbation is added to the distance matrix, as the level of noise increases.

LMDSE involves two parameters, including the number of landmark points, l, and the number of GCPs, p. In this experiment, we used $l = 100$ and $p = 100$. Fig. 4 shows the embedding solutions determined by classical MDS, LMDS, and LMDSE in cases that noise levels are 0%, 10%, 30%, and 50%. As shown in Fig. 4, classical MDS and LMDSE provide embedding solutions that are insensitive to high level of noise, whereas LMDS solutions are severely degraded in cases of 30% and 50% noise. It should be noted that the LMDSE solution is comparable to the classical MDS solution, although only 100 landmark points are used.

3.2. Experiment 2: Similar Image Retrieval

Similar image retrieval is one of important tasks in managing large amounts of images, the goal of which is to search images in the database that are similar to a query image. In general, such a task is not easy since images contain objects involving different sizes, illumination, rotation, resolution, modification, and so on even if they are similar. Employing the scale invariant feature transform (SIFT) [5] that has been widely used in object recognition, we demonstrate the useful behavior of our LMDSE in the case where distances between SIFT features are corrupted by high level of noise.

The SIFT-based algorithm for similar image retrieval is summarized below, which is a slight modification of the object recognition method that was used in [5]. We stress the useful behavior of our LMDSE as a metric-preserving dimensionality reduction, rather than presenting a new method for similar image retrieval.

Algorithm: Similar Image Retrieval

---

1. Compute SIFT features for given images. Each SIFT feature is a 128-dimensional vector and each image contains a large number of feature vectors. Denote by $\mathcal{F}_i = \{\boldsymbol{f}_1^i, \ldots, \boldsymbol{f}_{N_i}^i\}$ a set of SIFT feature vectors for image i, where N_i is the number of feature vectors and $\boldsymbol{f}_j^i \in \mathbb{R}^{128}$ for $j = 1, \ldots, N_i$.
2. Apply LMDS or LMDSE to reduce the dimension of SIFT feature vectors from 128 to 20.
3. We follow the method for image matching used in [5]. With abuse of notations, let $\mathcal{F}_i$ and $\mathcal{F}_j$ be two sets of dimension-reduced STFT feature vectors, corresponding to images i and j, respectively. Starting from $\boldsymbol{f}_1^i$ in $\mathcal{F}_i$, determine whether there is a $\boldsymbol{f}_k^j$ that matches $\boldsymbol{f}_1^i$ in the following way. We compute Euclidean distances between $\boldsymbol{f}_1^i$ and each element in $\mathcal{F}_j = \{\boldsymbol{f}_1^j, \ldots, \boldsymbol{f}_{N_j}^j\}$ to find feature vectors that are closest and 2nd closest to $\boldsymbol{f}_1^i$. If the ratio of the closest distance to the 2nd closest distance is less than a threshold set as 0.6 here, then $\boldsymbol{f}_1^i$ is considered to match the closest neighbor in $\mathcal{F}_j$, increasing the counter M_{ij} by one. Repeat this for $\boldsymbol{f}_2^i, \ldots, \boldsymbol{f}_{N_i}^i$.
4. Count the number of matched SIFT features between image i and j, which is stored in M_{ij}. If the following conditions are satisfied:

$$\frac{M_{ij}}{N_i} > 0.2 \text{ and } \frac{M_{ij}}{N_j} > 0.2,$$

then images i and j are determined to be similar.

In experiments, we used 124 thumbnail images that were collected on the World Wide Web by a search engine (see Fig. 5), consisting of 2-6 similar images for each. We compute 128-dimensional

14,173 SIFT feature vectors from these images, implying that a single image involves about 114 SIFT feature vectors on average. Classical MDS involves an eigendecomposition of a distance matrix $\boldsymbol{D} \in \mathbb{R}^{14173 \times 14173}$, which is not an easy task.

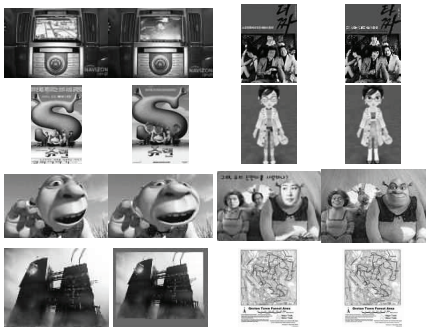

Fig. 5. Exemplary images are shown, where similar images are paired, each of which is slightly different.

We apply LMDS and LMDSE with 200 landmark points and 128 GCPs, to reduce the dimension of feature vectors from 128 to 20. Adding various levels of noise to the distance matrix as in Sec. 3.1, we compare the performance in terms of F-score, F, which is defined as,

$$F = 2 \cdot (\text{precision} \cdot \text{recall}) / (\text{precision} + \text{recall}).$$

The results are summarized in Fig. 7. As the noise level increases from 0% to 9%, F-score for LMDS decreases, while LMDSE provides robust performance for all noise levels. Furthermore, in cases of 10% or more noise, LMDS breaks down, where precision is not available (since there are no retrieved images) and recall is 0%.

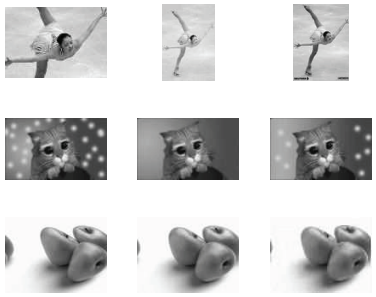

Fig. 6. Examples of similar images that are successfully retrieved by our method in the case of 10% noise. Retrieved similar images have subtle differences in illumination, size, and background pattern.

4. CONCLUSIONS

We have presented LMDSE where we combined individual LMDS solutions that operated on different partitions of the input, through affine transformations using GCPs. As shown in Fig. 3, LMDSE consists of 3 steps: (1) L individual LMDS solutions using $\boldsymbol{D}_i$'s; (2) affine transformations to align these individual LMDS solutions in a common coordinate system; (3) averaging aligned LMDS solutions. In contrast to LMDS, LMDSE used all of the input but in a piecewise manner so that the noise-robustness was much improved, with compatible scalability. Experiments on noisy grid and similar image retrieval demonstrated the noise-robustness behavior of LMDSE.

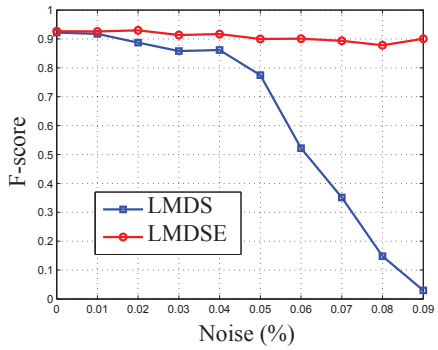

Fig. 7. F-scores are shown for LMDS and LMDSE, as noise level increases from 0% to 9%.

Acknowledgments: This work was supported by KOSEF Basic Research Program (Grant R01-2006-000-11142-0), National Core Research Center for Systems Bio-Dynamics, and KOSEF WCU Program (Project No. R31-2008-000-10100-0).

5. REFERENCES

[1] T. Cox and M. Cox, *Multidimensional Scaling*, 2nd ed. Chapman & Hall, 2001.

[2] V. de Silva and J. B. Tenenbaum, "Sparse multidimensional scaling using landmark points," Stanford University, Tech. Rep., 2004.

[3] C. Faloutsos and K. Lin, "FastMap: a fast algorithm for indexing, data-mining and visualization," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*, 1995, pp. 163–174.

[4] S. Ketpreechasawat and O. C. Jenkins, "Hierarchical landmark charting," Master's thesis, Brown University, 2006.

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[6] J. C. Platt, "FastMap, MetricMap, and landmark MDS are all Nyström algorithms," in *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2005, pp. 261–268.

[7] J. T. L. Wang, X. Wang, K. I. Lin, D. Shasha, B. A. Shapiro, and K. Zhang, "Evaluating a class of distance-mapping algorithms for data mining and clustering," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 1999, pp. 307–311.

[8] T. Yang, J. Liu, L. McMillan, and W. Wang, "A fast approximation to multidimensional scaling," in *Proceedings of the ECCV-2006 Workshop on Computation Intensive Methods for Computer Vision*, 2006.