Dimensionality Reduction for Image Retrieval

P. Wu, B.S. Manjunath and H.D. Shin
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106-9560
{peng, manj, hdshin}@iplab.ece.ucsb.edu

Abstract

Dimensionality reduction methods are of interest in applications such as content based image and video retrieval. In large multimedia databases, it may not be practical to search through the entire database in order to retrieve the nearest neighbors of a query. Good data structures for similarity search and indexing are needed, and the existing data structures do not scale well for the high dimensional multimedia descriptors. We investigate the use of weighted multi-dimensional scaling (WMDS) for dimensionality reduction. The main objective of the WMDS is to preserve the local topology of the high dimensional space, i.e., to map the nearest neighbors in the high dimensional space to nearest neighbors in the lower dimensional space. In addition to the well known retrieval accuracy as a measure of performance, we propose two additional measures that take into account the ordinal relationships among the nearest neighbors. Experimental results are given.

1 Introduction

The dimensionality of image descriptors (feature vectors) used in image retrieval applications, in general, is quite high. Typical descriptor dimensions range from few tens to several hundreds. For example, a color histogram may contain 256 bins. This high dimensionality of the feature vectors creates problems in constructing efficient data structures for search and retrieval. It is well known that most of the indexing structures do not scale well when the dimensionality of the feature vector exceeds 20 [11]. For this reason, there is considerable interest in reducing the dimensionality of the descriptors while preserving the original topology of the high dimensional space.

Previously investigated methods for dimensionality reduction include Principal Component Analysis (PCA) [9], Singular Value Decomposition (SVD) [5], Self-Organizing Map (SOM) [6], Fastmap [4] and Multidimensional Scaling (MDS) [2]. SOM is quite often used for the classification and clustering of the feature vectors to constrain the search space [7]. PCA or SVD [2] amount to rotating the coordinate axes of the high dimensional vector space so that projections onto the new axes result in uncorrelated feature points. Dimensionality reduction is achieved by using few of the rotated axes as basis vectors. In MDS, the low dimensional representation is found by minimizing certain cost functions. One example of such a cost function is the difference in pairwise distances between feature points in the original and lower dimensional space. In an ideal situation, for any given query one should find the same set of nearest neighbors in the lower dimensional space as in the original high dimensional space.

We propose a modified version of the MDS cost function that is motivated by the following observations in the context of image retrieval. (a) it is important to ensure that the top M retrievals (nearest neighbors) are preserved in the reduced dimensional space. M is typically a fraction of the entire database, and as such not all pairwise distances need be considered. (b) often the ordinal relationships of the retrievals are of much interest, and this relationship be preserved as well in the reduced dimension space. The proposed weighted MDS has two advantages over the traditional MDS. Unlike the traditional MDS, it requires much fewer data points (pairwise distances) to achieve the same level of performance as MDS. Further, it compares favorably with MDS in preserving the local topology, as our experimental results in Section 4 demonstrate.

In the next section we describe the weighted MDS approach. Section 3 discusses several performance metrics to evaluate the effectiveness of descriptors in a reduced dimensional space. Experimental results are given in Section 4 and concluding remarks in Section 5.

2 Weighted Multidimensional Scaling (WMDS)

Let $\Re_H$ denote the space of the high dimensional feature vectors. Let $\Re_L$ denote the reduced dimensional feature space. Let $F_H(i)$ and $F_L(i)$ represent the feature descriptors in the original high dimensional space and in the reduced lower dimensional space, respectively. Let $D_{ij} = dist(F_H(i), F_H(j))$ be the distance between two objects i and j in the original feature space. Let $d_{ij} = dist(F_L(i), F_L(j))$ be the corresponding distance in the lower dimensional space. We assume Euclidean distances between feature vectors. Then, the objective of weighted MDS (WMSD) is to minimize the cost function

$$E_{WMDS} = \frac{\sum_{i<j}^{N} \alpha_{ij}(d_{ij} - D_{ij})^2}{\sum_{i<j}^{N} D_{ij}^{\ 2}} \qquad (1)$$

where N is the total number of objects in the database and α_{ij} is a weighting factor associated with D_{ij}. For $\alpha_{ij} = 1$ for all i and j in (1), we have the traditional loss function defined for MDS

$$E_{MDS} = E_{WMDS}\big|_{\alpha_{ij} = 1} \qquad (2)$$

The minimization of (2) results in a configuration of N points in $\Re_L$. It is clear that the objective is to preserve the pairwise distances in the new reduced dimensional space. Note that since $D_{ij} = D_{ji}$ and $D_{ii} = 0$, among all N^2 pair distances, $(N^2 - N)/2$ of them are independent constraints to be satisfied in the minimization of (2). This loss function has been used in [3],[4], and different optimization methods have been investigated. We use the iterative majorization method [3] in our implementations for minimizing the E_{WMDS}.

The minimization techniques can be carried out even when partial information on the distances is available (see [3], Chapter 6). Figure 1 shows some results wherein only a partial set of pairwise distances are used. The performance is measured in terms of retrieval accuracy (see below). The pair distances are selected at random. Note that the performance using only 50% of the data is comparable with that of using the entire distance matrix.

This observation that partial data can be used in MDS leads us to the weighted MDS formulation wherein the distance constraint on the nearest neighbors is given priority. This is particularly appealing in the context of image retrieval in that we are typically interested only in the top few retrievals. Preserving the local topology of the higher dimensional space in the reduced dimensions is important. Thus, one can specify the weights α_{ij} in the WMDS to be inversely proportional to the D_{ij}s. Since the number of nearest neighbors considered is also of interest, we propose the following choices of α_{ij}

$$\alpha_{ij}(m) = \exp(-D_{ij}/T(i, m)) \qquad (3)$$

where $T(i, m)$ is the distance between an object i and its mth nearest neighbor. Since the value of $\alpha_{ij}(m)$ drops quite fast, one can further threshold its value to speed-up computations without affecting the quality of the results.

3 Performance Measures

Retrieval accuracy is often used as a measure of effectiveness of a descriptor in image retrieval applications. For evaluating the performance of the dimensionality reduction methods, we focus only on preserving the local topology of the high dimensional space and not on the perceptual similarity measure.

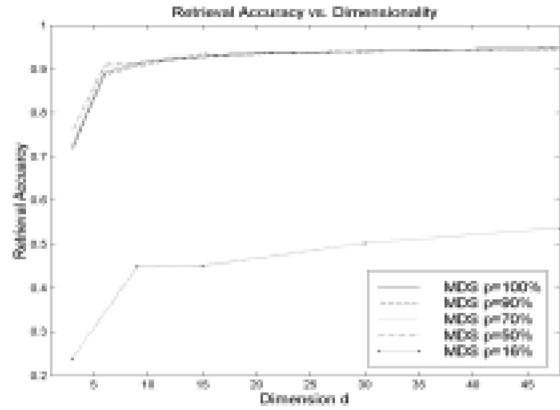


Fig 1. Retrieval accuracy using MDS and partial distances. “p” denotes the percentage of data used.

Notations: Let q denote a query object. Let $S^{(H)}(q) = \{s_1^h, s_2^h, \ldots, s_M^h\}$ be an ordered set of the first M nearest neighbors of q in $\Re_H$. The elements are arranged such that

$$dist(F_H(q), F_H(s_1^h)) < dist(F_H(q), F_H(s_2^h)) < \ldots$$

Similarly, let $S^{(L)}(q) = \{s_1^l, s_2^l, \ldots, s_M^l\}$ be the ordered set of the first M nearest neighbors of q in $\Re_L$. Let

$$S(q) = S^{(H)}(q) \cap S^{(L)}(q) = \{s_1, s_2, \ldots, s_K\}$$

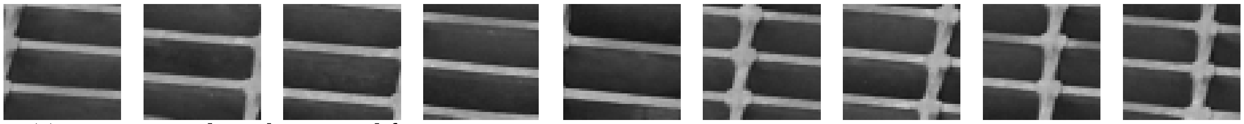
where $dist(F_L(q), F_L(s_1)) < dist(F_L(q), F_L(s_2)) < \ldots$.

Let $S(q)' = \{s_1', s_2', \ldots, s_K'\}$ be a permutation of $S(q)$, in which the element are arranged such that

$$dist(F_H(q), F_H(s_1')) < dist(F_H(q), F_H(s_2')) < \ldots$$

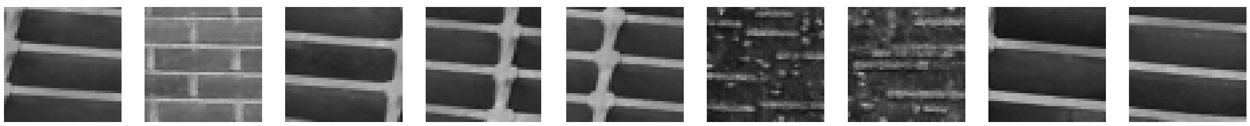
Let $K(q) = |S(q)|$, the cardinality of $S(q)$. Then the retrieval accuracy for a query object q is computed as $K(q)/M$.

Retrieval accuracy alone is not a sufficient measure of performance. Often the order in which the items are retrieved is of equal importance. Consider Figure 2(b) and Figure 2(c). Both correspond to retrievals using the first picture in each row as the query, and the retrieval rates are the same for both cases. However, if the users are asked to rank the performances, results in Figure 2(b) are perceptually more appealing than the results in Figure 2(c). One reason being that perceptually more similar items appear farther away in the rank-ordered list in Figure 2(c). If we assume that the ground truth corresponds to the nearest neighbors in $\Re_H$, then one can impose an additional constraint that requires maintaining the ordinal relationships in the reduced dimensional space $\Re_L$.



(a) top 8 retrievals in the original feature space

(b) top 8 retrievals in the reduced dimensional space (m=100): $K/M = 0.625$, $\beta = 0.69$, $\eta = 0.33$, $\beta/\eta = 2.09$



(c) top 8 retrievals in the reduced dimensional space (m=50): $K/M = 0.625$, $\beta = 0.66$, $\eta = 0.66$, $\beta/\eta = 1.0$

Fig 2. The left most image in each row is the query image.

It may help to consider another simple example wherein a query q has the top 5 nearest neighbors as $\{A, B, C, D, E\}$ in $\Re_H$. Consider two sets of nearest neighbors in reduced dimensional space: $L_1 = \{x, A, B, C, x\}$ and $L_2 = \{x, x, C, D, E\}$. Here "x" represent items not in the ground truth. Both L_1 and L_2 have K=3 retrievals from the ground truth (corresponding to 60% retrieval rate) but clearly the first set L_1 preserves the topology better for the query q than the second set L_2.

For a given set $S = \{s_1, s_2, \ldots, s_K\}$, for $s \in S$, define

$$rank(s, S) = i \qquad if\ s = s_i \tag{4}$$

For example, $rank(A, L_1) = 2$ and $rank(D, L_2) = 4$. A normalized measure based on the rank of the retrieved nearest neighbors can be defined as:

$$\beta(q) = \frac{2}{M(M+1)}\left[(M+1)K(q) - \sum_{s_i \in S(q)} rank(s_i, S^{(H)}(q))\right] \tag{5}$$

$\beta(q)$ is normalized to [0,1], with a higher value indicating a better retrieval set. For the two examples L_1 and L_2 mentioned in the previous paragraph, the values of $\beta(q)$ are 0.8 and 0.4, respectively, even though both have the same retrieval accuracy. However, this still does not take into account the relative positions in which the objects appear. Thus $\{x, A, B, C, x\}$ and $\{x, C, B, A, x\}$ both have the same β.

To address this, in addition to the individual ranks of the retrieved objects, we need to consider their ordinal relationships as well. One objective measure is to compare the rank of the object in $\Re_L$ to its rank in $\Re_H$, as in the following:

$$\eta(q) = \frac{1}{W(q)} \sum_{s_i \in S(q)} \left| rank(s_i, S(q)) - rank(s_i, S(q)') \right| \tag{6}$$

where $W(q)$ is the maximum possible distance between two permutations and is given by

$$W(q) = \begin{cases} K(q)^2/2 & K(q) = even \\ (K(q)^2 - 1)/2 & K(q) = odd \end{cases} \tag{7}$$

$\eta(q)$ is also normalized to [0,1], with a smaller $\eta(q)$ indicating a better retrieval. $\beta(q)$ together with $\eta(q)$ can be used to evaluate the effectiveness of the descriptors if the ground truth for each query is know. When the β for two sets of retrieval are about the same, η provides a quantitative measure that can be used to rank order the effectiveness. In Figure 2, the β for retrievals in (b) (β =0.69) is about the same as that for (c) (β = 0.65), but η captures the relative ordering quite well. The η for (b) is 0.33, only half as much as the η for (c), which is 0.66. Perhaps a good overall effectiveness measure is the ratio $\beta(q)/\eta(q)$.

We note that the average retrieval rate defined in the MPEG-7 core experiments [12] is very similar to the $\beta(q)$ defined above (except that the optimal value of the average retrieval rate is "0" in [12] and a higher value indicates a poorer performance of the descriptors.)

4 Experimental Results

A texture image database collected from MPEG-7 evaluation data set is chosen as the test data set [12]. It consists of 832 real world texture images from 52 different texture classes and each of them contains 16 texture images. We use the Gabor texture features described in [8] as our image descriptors. In [8], a given image is filtered using scale and orientation sensitive filters modeled using Gabor functions. The mean and standard deviations of the filtered outputs are used to construct the texture descriptor. We use a total of 24 filters, thus creating a 48-dimensional texture descriptor for each image in the database.

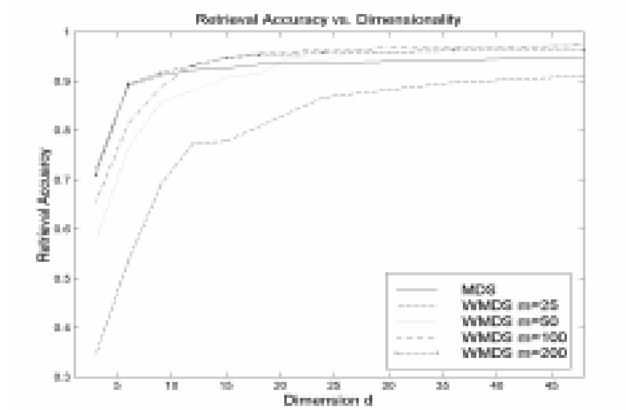


Fig 3. Retrieval accuracy using WMDS. “m” denotes the number of top nearest neighbors considered, as specified in (3)

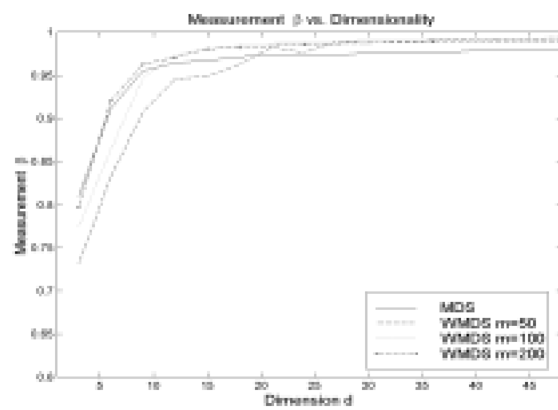


Fig 4. Using β as a performance measure

In evaluating the performance of dimensionality reduction, the nearest neighbors for each image object in the high dimensional space is used as the ground truth. Since each image has 16 textures from the same class, we set M=16 in computing the retrieval accuracy.

Figure 3 gives an evaluation of the reduced dimensional descriptors obtained using the WMDS method. Different sets of nearest neighbors are considered, corresponding to m=25, 50, 100, and 200 in (3). Note that the performance for m=50 is within 10% of the full dimensional MDS. In terms of the number of pairwise distances used in computing the mapping, this is comparable to p=16% in Figure 1. For p=16% of data, MDS method performs significantly worse. However, 16% of the data points is randomly selected in the experiments. It is interesting to note that the WMDS with m=50 outperforms the full MDS when the reduced dimensionality exceeds 20. For values of m much less than 50, the performance of the WMDS is not good, as is evident by the curve for m=25 in Figure 3.

Figure 4 plots the β as a function of number of dimensions (of $\Re_L$) for WMDS and MDS. We observe that m=50 offers a good trade-off between computational complexity and performance.

5 Conclusions

We have presented a simple but effective method for dimensionality reduction using weighted MDS. On a texture image dataset, we have demonstrated that using a fraction of the nearest neighbors, one can compute a lower dimensional representation that is quite effective. This method compares favorably in terms of computational complexity with the full MDS method. We are currently evaluating and comparing the performance of WMDS with other dimensionality reduction methods such as the FastMap and SVD. One issue that has not been addressed in this work is the mapping of new feature vectors onto the lower dimensional space, and different techniques for computing this transformation are being investigated.

Acknowledgments: We would like to thank Professor S. Chandrasekaran at UCSB for many useful discussions. This research is supported in part by grants from Samsung Electronics and the Institute for Scientific Computing Research (ISCR) at LLNL.

References

[1] S. D. Backer, A. Nard and P. Scheunders, “Non-linear dimensionality reduction techniques for unsupervised feature extraction,” Pattern Recognition Letters 19, pp. 711-720, 1998.
[2] M. Beatty and B. S. Manjunath, “Dimensionality reduction using multidimensional scaling for image search,” Proc. IEEE ICIP, Santa Barbara, CA, Vol. II, pp. 835-838, Oct. 1997.
[3] I. Borg, P. J. F. Groenen, “Modern Multidiemsional Scaling,” Springer, New York, 1997.
[4] C. Faloutsos and K. Lin, “Fastmap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets,” Technical Report CS-TR-3383, Univ. of Maryland Institute for Advanced Computer Studies, Jan. 1994.
[5] D. Hull, “Improving text retrieval for the routing problem using latent semantic index,” Proc. of the 17th ACM-SIGIR Conference, pp. 282-291, 1994.
[6] T. Kohonen, “Sef-Organizing Maps,” Springer, Berlin, 1995.
[7] W.Y. Ma and B.S. Manjunath, "Texture features and learning similarity," Proc. Of IEEE CVPR’96, pp. 425-430, San Francisco, CA, June 1996.
[8] B. S. Manjunath and W. Y. Ma, “Texture features for browsing and retrieval of image data,” IEEE Trans. on Pattern Analysis and Machine Intelligence, 18(8), pp. 837-842, 1996.
[9] R. Ng and A. Sedighian, “Evaluating multi-dimensional indexing structures for images transformed by principle component analysis,” Proc. of the SPIE, 2670, pp. 50-61, 1994.
[10] A. R. Webb, “Multidimensional scaling by iterative majorization using radial basis functions,” Pattern Recognition, Vol. 28, No. 5, pp. 753-759, 1995.
[11] D. White and R. Jain, “Similarity indexing with the SS-tree,” Proc. Int. Conf. on Data Engineering, pp. 516-523, 1996.
[12] MPEG Document #2929, Description of Color/Texture core experiments, October 1999, Melbourne, Australia.