Face Recognition Experiments with Random Projection

Navin Goel[a], George Bebis[a], and Ara Nefian[b]

[a]Computer Vision Laboratory, University of Nevada, Reno
[b]Future Platforms Department, Intel Corporation, Santa Clara

ABSTRACT

There has been a strong trend lately in face processing research away from geometric models towards appearance models. Appearance-based methods employ dimensionality reduction to represent faces more compactly in a low-dimensional subspace which is found by optimizing certain criteria. The most popular appearance-based method is the method of eigenfaces that uses Principal Component Analysis (PCA) to represent faces in a low-dimensional subspace spanned by the eigenvectors of the covariance matrix of the data corresponding to the largest eigenvalues (i.e., directions of maximum variance). Recently, Random Projection (RP) has emerged as a powerful method for dimensionality reduction. It represents a computationally simple and efficient method that preserves the structure of the data without introducing significant distortion. Despite its simplicity, RP has promising theoretical properties that make it an attractive tool for dimensionality reduction. Our focus in this paper is on investigating the feasibility of RP for face recognition. In this context, we have performed a large number of experiments using three popular face databases and comparisons using PCA. Our experimental results illustrate that although RP represents faces in a random, low-dimensional subspace, its overall performance is comparable to that of PCA while having lower computational requirements and being data independent.

Keywords: Face Recognition, Random Projection, Principal Component Analysis

1. INTRODUCTION

Considerable progress has been made in face recognition research over the last decade, especially with the development of powerful models of face appearance.[1] These models represent faces as points in high-dimensional image spaces and employ dimensionality reduction to find a more meaningful representation, therefore, addressing the issue of the "curse of dimensionality".[2] The key observation is that although face images can be regarded as points in a high-dimensional space, they often lie on a manifold (i.e., subspace) of much lower dimensionality, embedded in the high-dimensional image space.[3] The main issue is how to properly define and determine a low-dimensional subspace of face appearance in a high-dimensional image space.

Dimensionality reduction techniques using linear transformations have been very popular in determining the intrinsic dimensionality of the manifold as well as extracting its principal directions (i.e., basis vectors). The most prominent method in this category is PCA.[2] PCA determines the basis vectors by finding the directions of maximum variance in the data and it is optimal in the sense that it minimizes the error between the original image and the one reconstructed from its low-dimensional representation. PCA has been very popular in face recognition, especially with the development of the method of eigenfaces.[4] Its success has triggered significant research in the area of face recognition and many powerful dimensionality reduction techniques (e.g., Probabilistic PCA, Linear Discriminant Analysis (LDA) Independent Component Analysis (ICA), Local Feature Analysis (LFA), Kernel PCA) have been proposed for finding appropriate low-dimensional face representations.[1]

Recently, RP has emerged as a powerful dimensionality reduction method.[5] Its most important property is that it is a general data reduction method. In RP, the original high-dimensional data is projected onto a low-dimensional subspace using a random matrix whose columns have unit length. In contrast to other methods, such as PCA, that compute a low-dimensional subspace by optimizing certain criteria (e.g., PCA finds a subspace that maximizes the variance in the data), RP does not use such criteria, therefore, it is data independent. Moreover, it represents a computationally simple and efficient method that preserves the structure of the data without introducing significant distortion. For example, there exist a number of theoretical results supporting that RP

Email: (goel,bebis)@cs.unr.edu, ara.nefian@intel.com

preserves approximately pairwise distances of points in Euclidean space,[6] volumes and affine distances,[7] and the structure of data (e.g., clustering).[5]

RP has been applied on various types of problems yielding promising results (see Section 2.2 for a brief review). In this paper, our goal is to investigate the feasibility of RP for face recognition. Specifically, we have evaluated RP for face recognition under various conditions and assumptions, and have compared its performance to PCA. Our results indicate that RP compares quite favorably with PCA while, at the same time, being simpler, more computationally efficient, and data independent. Our results are consistent with previous studies comparing RP with PCA,[5,6,8,9] indicating that RP might be an attractive alternative for dimensionality reduction in certain face recognition applications.

The rest of the paper is organized as follows: in Section 2 we review RP and present a brief review of its properties and applications. In Section 3, we discuss using RP for face recognition and present the main steps of such an approach. Section 4 presents our experiments and results. Finally, Section 5 contains our conclusions and directions for future research.

2. RANDOM PROJECTION

RP is a simple yet powerful dimension reduction technique that uses random projection matrices to project data into low-dimensional spaces. We summarize below the key points of RP and present a brief review of its applications.

2.1. Background

Let us suppose that we are given a set of vectors Γ_i, $i = 1, 2, ..., M$, in a p-dimensional space. Then, RP transforms Γ_i to a lower dimension d, with $d << p$, via the following transformation:

$$\widetilde{\Gamma_i} = R\Gamma_i \tag{1}$$

where R is orthonormal and its columns are realizations of independent and identically distributed (i.i.d.) zero-mean normal variables, scaled to have unit length.

RP is motivated by the *Johnson-Lindenstrauss* lemma[10] that states that a set of M points in a high-dimensional Euclidean space can be mapped down onto a $d >= O(logM/\epsilon^2)$ dimensional subspace such that the distances between the points are approximately preserved (i.e., not distorted more than a factor of $1 \pm \epsilon$, for any $0 < \epsilon < 1$). A proof of this theorem as well as tighter bounds on ϵ and d are given in[11] and.[12] There is also experimental evidence suggesting that the lower bound for d is not very tight and that, in practice, lower d values have yielded good results.[13]

The entries of the matrix R can be calculated using the following algorithm[5]:

Step 1. Set each entry of the matrix to an i.i.d. $N(0, 1)$ value.

Step 2. Orthogonalize the d rows of the matrix using the Gram-Schmidt algorithm.

Step 3. Normalize the rows of the matrix to unit length (i.e., important for preserving similarities in the low-dimensional space).

The main reason for orthogonalizing the random vectors is to preserve the similarities between the original vectors in the low-dimensional space. In high enough dimensions, however, it is possible to save computation time by avoiding the orthogonalization step without affecting much the quality of the projection matrix. This is due to the fact that, in high-dimensional spaces, there exist a much larger number of almost orthogonal vectors than orthogonal vectors.[14] Thus, high-dimensional vectors having random directions are very likely to be close to orthogonal.

It should be mentioned that a simpler algorithm has been proposed by Achlioptas[12] for approximating the random matrix, yielding significant computational savings during the compution of R and the projections $R\Gamma_i$.

The algorithm produces a sparse random matrix with elements belonging in -1, 0, 1, therefore, simplifying the computations considerably. Brigham et al.[8] has found that this algorithm performs quite in practice. In this study, we have chosen to compute the random matrix using the 3-step algorithm outlined above to avoid introducing potential errors in our results due to the non-orthogonality of R.

2.2. Applications

RP has been applied on various types of problems including information retrieval,[15, 16] machine learning,[5, 6, 8, 9, 17, 18] and optimization.[19] Although it is based on a simple idea, RP has demonstrated good performance in a number of applications, yielding results comparable to conventional dimensionality reduction techniques, such as PCA, while having much lower computational requirements.

Several researchers have introduced RP in nearest-neighbor query problems which arise quite often in database applications when performing similarity searching. Kleinberg[15] developed an algorithm for finding approximate nearest neighbors by combining randomly chosen one-dimensional projections of the underlying data. Indyk and Motwanti[20] used RP, in the form of a locality-sensitive hashing, as part of a randomized algorithm for solving the nearest-neighbor problem in high dimensions. Using random projection to reduce the original problem to a series of tractable low-dimensional problems, they produced an algorithm that scales better with dimension that deterministic methods and is efficiently implementable in practice for sparse indexing of databases.

Papadimitriou et al.,[21] have combined RP with Latent Semantic Indexing (LSI) for document categorization and classification. LSI has shown to be an elegant and accurate technique for solving these problems, however, its high computational cost make it infeasible for large databases. In their approach, RP was used to speed-up LSI by reducing the dimensionality of the data prior to applying LSI. In a similar manner, Kurimo[22] applied RP to the indexing of audio documents, prior to using LSI and Self-Organizing Maps (SOMs).

Dasgupta[5] combined RP with Expectation-Maximization (EM) to learn high-dimensional Gaussian mixture models. His results illustrate that data from a mixture of k Gaussians can be projected down to $O(logk)$ dimensions while retaining data separation. He applied this algorithm on a hand-written digit data set achieving good results. Recently, Li et al.[18] used RP with EM to learn geometric object appearance for object recognition. Motivated by the results of,[5] Fern and Brodley[17] investigated the application of RP for clustering high-dimensional data. Using single runs of RP, they demonstrated that the clustering results were very unstable. To deal with this problem, they proposed using ensembles of RP. Using three different data sets, they demonstrated better results than using individual runs of RP but also clustering using the EM algorithm and PCA for dimensionality reduction.

Kaski,[6] presented experimental results using RP in the context of a system for organizing textual documents using SOMs (i.e., WEBSOM). His results illustrate that RP needs moderate number of dimensions for producing a good mapping. In this case, the results were as good as those obtained using PCA, and almost as good as those obtained using the original vectors. Brigham and Manilla[8] compared several dimensionality reduction techniques on image and text data. Their results indicate again that RP preserves distances and has performance comparable to that of PCA while being much faster. More recently, Fradkin and Madigan,[9] evaluated RP in the context of supervised learning. In particular, RP was compared with PCA on a number of different problems using different machine learning algorithms. They concluded that although RP was slightly worst than PCA, its computational advantages might make it attractive in certain applications.

Other applications of RP include solving VLSI layout,[19] performing approximate kernel computations,[23] similarity computations for histogram models,[16] protein similarity search,[24] and DNA motif discovery.[25]

3. FACE RECOGNITION USING RANDOM PROJECTION

There are several reasons that make RP attractive for face recognition. First of all, RP is data independent. This is in contrast to other techniques such as PCA, where the data is projected on a low-dimensional subspace computed from a training set. This implies that RP is faster to compute compared to PCA as discussed in the next paragraph. In particular, PCA becomes very expensive to compute when the dimensionality of the data is very high. Although there exist efficient algorithms for computing the largest eigenvectors (i.e., the ones corresponding to the largest eigenvalues) without considering the original covariance matrix which is very large (see[4]), the number of eigenvectors computed by these methods is limited by the number of training data. This implies that these algorithms might not be very useful when the data set is small.

Second, RP does not require that the face representations are updated when the face database changes (i.e., when new faces are added to the database or old faces are deleted). In contrast, traditional dimensionality reduction techniques, such as PCA, require recomputing the basis vectors and re-projecting the faces for maintaining their optimality. Our experimental results in Section 4 (i.e., subjects with same/different identity in the training/gallery sets) illustrate this fact for PCA. Although there exist algorithms for updating the basis vectors adaptively,[26] there are stability issues with algorithms while the faces still have to be re-projected on the updated basis.

Finally, RP has much lower computational requirements compared to PCA. Computing a $d \times p$ random matrix is $O(pd^2)$[5] using the algorithm presented in Section 2.1 and $O(pd)$ using the simplified algorithm by Achlioptas.[12] On the other hand, traditional methods such as PCA and SVD are more expensive with PCA being $O(p^2M)+O(p^3)$ and SVD being $O(prM)$ assuming sparse matrices with r non-zero entries per column.[21, 27] It should be mentioned that in general, performing the projection of M points is $O(pdM)$ both for PCA and RP. However, when approximating the projection matrix using integer values as in,[12] the projections can be computed much faster.

RP has shown to have several other properties that can benefit face recognition. For example, it has been shown that when the original data form highly eccentric Gaussian clusters, RP preserves the clustering effect down in lower dimensions (i.e., up to $O(logk)$ dimensions where k is the number of clusters) and produces more spherical clusters[5] which could improve recognition performance. In addition, it has been shown that for moderate dimensions, the inner product between the mapped vectors follows closely the inner product of the original vectors, that is, random mappings do not introduce distortion in the data and preserve similarities approximately.[27]

Below, we outline the steps of face recognition using RP. The algorithm is similar to the eigenface approach except that the projection matrix is computed randomly instead of applying PCA on the sample covariance matrix of the data. Representing each image I as a $N \times N$ vector Γ_i, we follow the steps below to represent I in a low-dimensional random space:

Step 1. Compute the average face:

$$\Psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i \tag{2}$$

where, M is the number of face images in the training set.

Step 2. Subtract the mean face from each other face:

$$\Phi_i = \Gamma_i - \Psi \tag{3}$$

Step 3. Generate the random matrix using the algorithm given in Section 2.1.

Step 4. Project the normalized face images in the random subspace:

$$w_i = R\Gamma_i \tag{4}$$

Each face in the training set is then represented by the coefficients of projection w_i. During recognition, the input face is normalized the same way and projected in the same random space. Then, the face is recognized by comparing its projections to the projections of the training images, using a similarity-based procedure. Three different procedures have been considered in this study (see Section 4.2).

4. EXPERIMENTS AND RESULTS

In this section, we evaluate RP for face recognition by conducting a large number of experiments using different scenarios and several data sets.

4.1. Data Sets

We have considered three popular face databases in this study: ORL,[28] CVL,[29] and AR.[30] The ORL database[28] contains 400 face images from 40 subjects with 10 frontal exposures of each assuming different facial expressions, lighting, and slight orientation changes. The CVL database[29] contains 798 images from 114 subjects with 7 exposures of each assuming different facial expressions and orientations. We only used the frontal exposures of each subject in our experiments (i.e., 3 exposures assuming different facial expressions, that is, 342 images). The AR database[30] is the most extensive of all, containing 3,276 images from 126 subjects. There are 26 frontal exposures for each subject taken in two different sessions assuming neutral poses (2 images), different facial expressions (6 images), illumination changes (6 images), occlusions due to sunglasses (6 images) and occlusions due to scarf (6 images). In each experiment, we divided the database under experimentation into three subsets: training, gallery, and test. The training set was used only in the case of PCA (i.e., to compute the eigenspace), which was considered in this study for comparison purposes.

4.2. Experimental Procedure

For each database, we have performed a number of experiments using three different procedures for evaluating recognition accuracy (i) closest match, (ii) majority voting, and (iii) scoring.

Given a test face, the closest match approach performs recognition by finding the nearest face from the gallery set. In this case, recognition accuracy is computed as the ratio of the faces recognized correctly from the test set over the total number of faces in the test set. To account for instabilities in the performance of RP due to the random nature of the projection matrix, the recognition performance reported for RP is the average over five different RPs.

In the majority voting approach, we run five different RPs (i.e., the number of RPs was chosen arbitrarily; running more RPs would produce more stable results) for each test face and find the closest match in each case. Then, the identity of the face is determined by applying majority voting on the five closest matches found. Recognition accuracy is computed by dividing the number of correctly recognized faces from the test set with the total number of faces in the test set.

Given a test face, the scoring approach retrieves the closest x matches from the gallery set. Then, we compute the recall rate for each test face, that is, the ratio of the face images retrieved from the gallery set that belong to the same person as in the test face, over the total number of images stored in the gallery for that person. Recognition accuracy is computed by averaging the recall rates for the test set. In the case of RP, we report again the average over five different RPs to account for instabilities in the performance of RP due to the random nature of the projection matrix.

In each experiment, we have evaluated and compared RP with PCA by varying the number of dimensions as well as the number of images per subject in the gallery set. We have also experimented with varying the identity of the subjects in the training and gallery sets. This has an effect only on the performance of PCA since RP is independent of the training set. We describe below our experiments and results in more detail.

4.3. Recognition Results Using Closest Match

First, we tested RP and PCA using the closest match approach. Given a test face, we used RP to project it in a random, low-dimensional, space using the algorithm outlined in Section 3. The same procedure was used for PCA except that the low-dimensional space was built using the training set. For each experiment, we varied the number of dimensions and reported the recognition accuracy using the three procedures discussed in Section 4.2.

We have performed two types of experiments: (a) the training set contains images of the same subjects as in the gallery set, and (b) the training and gallery sets contain different subjects. The second scenario is more realistic when we do not have a representative set of images for training (e.g., we do not have enough subjects or we do not have enough images under different illumination conditions). In this case, we would need to use images of other subjects to create a representative training set and compute the eigenspace.

Figures 1(a)-(c) show the results using the ORL database assuming subjects with the same identity both in the training and gallery sets while Figures 1(d)-(f) show the case of subject having different identity. In the first case, we built the training set using 3 images from each subject (i.e., 120 images) while in the second case, we built the training set using the images of 13 subjects (i.e., 130 images). The images of the rest 27 subjects (i.e., 270 images) were used to create the gallery and test sets.

To test the sensitivity of each method, we also varied the number of images per subject in the gallery set, (i.e., Figure 1(a) - 3 images per subject, Figure 1(b) - 2 images per subject, Figure 1(c) - 1 image per subject, Figure 1(d) - 5 images per subject, Figure 1(e) - 4 images per subject, and Figure 1(f) - 3 images per subject). In each graph, the blue line corresponds to PCA while the red line corresponds to RP.

Our first observation is that PCA in general performs better when the identity of the subjects in the gallery set is the same to the identity of the subjects in the training set. Comparing RP with PCA, our results show that PCA performs better than RP mostly for low dimensions (i.e., 20-30). This result is consistent with previous studies where it has been reported that RP compares favorably with PCA for moderate or higher number of dimensions. The difference in performance becomes smaller and smaller as the number of dimensions increases.

The results using different subjects in the gallery set than in the training set are more interesting. In this case, it is obvious that RP and PCA have very close performances. In fact, RP seems to be doing slightly better (i.e., 1%-2%) than PCA for higher dimensions. This is mainly because PCA is data dependent while RP is data independent. Overall, both methods seem to be affected by the number of images per subject in the gallery set. Their performance degrades as the number of images per subject in the gallery set decreases. The degradation in the performance of the two methods has been emphasized by the fact we increase the size of the test set as we decrease the size of the gallery set (i.e., the images removed from the gallery set are added to the test set).

The results for the CVL database are shown in Figure 2. Figure 2(a) corresponds to using subjects with the same identity both in the training and gallery sets while Figures 2(b) and (c) correspond to different subjects. In Figure 2(a), we built the training, gallery, and test sets by splitting the CVL database into three equal parts, keeping 1 image per subject for each part. In Figures 2(b) and (c) we built the training set using the images of 34 subjects (i.e., 102 images). Figure 2(b) corresponds to storing 2 images per subject in the gallery set while Figures 2(c) corresponds to storing 1 image per subject in the gallery set. The results for the CVL database are quite consistent with those for the ORL database. Figure 2(b) presents an interesting case where RP seems to be doing better than PCA (i.e., 3%-5%) using 50 dimensions and higher.

4.4. Recognition Results Using Majority Voting

It has been reported that RP performs better when averaging the results over several RPs.[17] Motivated by this observation, we report in this section experiments using majority voting on a relatively small ensemble of five different random projections. The results for the ORL and CVL databases are shown using the green line in Figures 1 and 2. Since we have performed the same exactly experiments as in the case of the closest match approach, we chose to plot the results on the same graphs to facilitate comparisons. Our results indicate that there is a slight improvement in performance using majority voting. In some cases, (i.e., Figure 2(b)), the improvement seems to be more important than in others.

Figure 3 shows the results using the AR database and majority voting. The blue line corresponds to PCA while the red line corresponds to RP. In these experiments, the training set was built by choosing 300 images randomly from the whole AR data set. In Figure 3(a), the gallery set was built using the neutral poses of each subject while the test set was built using the frames corresponding to different facial expressions. In Figure 3(b), we used the neutral poses for the gallery set and the frames corresponding to illumination changes for testing. In Figure 3(c) and (d), we built the gallery set using both the neutral poses and the frames corresponding to illumination changes. For testing, we used the frames with sunglasses (i.e., Figure 3(c)) and the frames with the scarf (i.e., Figure 3(d)). Our results indicate that RP using majority voting and PCA have similar performances with PCA doing better for lower dimensions and RP doing better for higher dimensions. The only exception is when using occlusion due to scarf for testing in Figure 3(d) where PCA seems to perform better for dimensions up to 200. As expected, the performance of both methods degrades seriously with occlusion due to sunglasses and scarf.

4.5. Recognition Results Using Scoring

In this set of experiments, we tested RP and PCA using scoring. It should be mentioned that scoring yields lower accuracy in general compared to the closest match and majority voting due to the fact that it evaluates the recall ratio as discussed in Section 4.2. The results for the ORL database are shown in Figure 4, for the CVL database in Figure 5, and for the AR database in Figure 6. The number of images x retrieved in each case is shown in the corresponding graphs. When the subjects in the training and gallery sets have the same identity, PCA shows better performance as before, however, differences in performance decrease with increasing the number of dimensions. When the identity of the subjects is different, then both methods have yielded comparable performances. In the case of the AR database, the performance drops significantly when testing both methods on occlusion due to sunglasses and scarf.

5. CONCLUSIONS

We have presented an experimental study to evaluate RP for face recognition. Our results indicate that RP compares favorably with PCA, especially when using majority voting and the training and gallery sets contain subjects having different identity. However, RP is much faster to compute and data independent which might be important factors to consider in a given face recognition application.

For future research, we plan to investigate more systematically the use of ensembles of RPs for face recognition. Overall, we did not obtain significant performance improvements in our experiments using majority voting. This could be due to the small ensemble size or the simplicity of the rule for combining the decisions. We plan to investigate much larger ensembles as well as more powerful combination rules such as those presented in.[31]

Using ensembles of RPs is a very promising idea since they can be built very fast and cheap. Preliminary results using ensembles of size 30 have shown that there is good diversity in recognition performance among different RPs which is important for improving overall performance. We also plan to perform comparisons with more powerful dimensionality reduction techniques such as LDA and ICA.

REFERENCES

1. W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys* **35, no. 4**, pp. 399–458, 2003.
2. R. Duda, P. Hart, and D. Stork, *Pattern Classification*, Jon-Wiley, 2nd edition, 2001.
3. B. Moghaddam, "Principal manifolds and probabilistic subspaces for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24, no. 6**, pp. 780–788, 2002.
4. M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience* **3**, pp. 71–86, 1991.
5. S. Dasgupta, "Experiments with random projection," in *Uncertainty in Artificial Intelligence*, 2000.
6. S. Kaski, *Data exploration using self-organizing maps*, cta Polytechnica Scandinavica, Mathematics, Computing and Management in engineering Series, 1997.
7. A. Magen, "Dimensionality reductions that preserve volumes and distance to affine spaces, and their algorithmic applications," in *6th. International Workshop on Randomization and Approximation Techniques in Computer Science*, 2002.
8. E. Brigham and H. Maninila, "Random projection in dimensionality reduction: applications to image and text data," in *International Conference on Knowledge Discovery and Data Mining*, pp. 245–250, 2001.
9. D. Fradkin and D. Madigan, "Experiments with random projection for machine learning," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 571–522, 2003.
10. W. Johnson and J. Lindenstrauss, "Extensions of lipschitz mapping into a hilbert space," in *Conference on Modern Analysis and Probability*, pp. 189–206, 1984.
11. S. Dasgupta and A. Gupta, "An elementary proof of the johnson-lindenstrauss lemma," in *UTechnical Report TR-99-006, International Computer Science Institute*, Berkeley, CA, 1999.
12. D. Achlioptas, "Database-friendly random projections," in *ACM Symposium on the Principles of Database Systems*, pp. 274–281, 2001.

13. E. Bingham and H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," in *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 245–250, 2001.
14. R. Hecht-Nielsen, "Context vectors: General purpose approximate meaning representations self-organized from raw data," in *Computational Intelligence: Imitating Life (Zurada et al. eds.)*, pp. 43–56, 1994.
15. J.M. Kleinberg, "Two algorithms for nearest-neighbor search in higher dimensions," in *Proceedings of 29th ACM Symp. on Theory of Computing*, pp. 599–608, 1997.
16. P. Thaper, S. Guha, and N. Koudas, "Dynamic multidimensional histograms," in *ACM SIGMOD*, pp. 428–439, 2002.
17. X.Z. Fern and C.E. Brodley, "Random projection for high dimensional data clustering: A cluster ensemble approach," in *Proceedings of the Twentieth International Conference on Machine Learning*, 2003.
18. W. Li, G. Bebis, and N. Bourbakis, "Integrating algebraic functions of views with indexing and learning for 3ọbject recognition," in *IEEE Workshop on Learning in Computer Vision and Pattern Recognition*, 2004.
19. S. Vempala, "Random projection: A new approach to vlsi layout," in *IEEE 39th Annual Symposium on Foundations of Computer Science*, pp. 389–395, 1998.
20. P. Indyk and R. Motwani, "Appropriate nearest neighbors: towards removing the curse of dimensionality," in *Proceedings of 30th ACM Symp. on Theory of Computing*, pp. 604–613, 1998.
21. C.H. Papadimitriou, P. Raghvan, H. Tamaki and S. Vempala, "Latent semantic analysis: A probabilistic analysis," in *Proceedings of 17th ACM Symp. On the principles of Database Systems*, pp. 159–168, 1998.
22. M. Kurimo, *Indexing audio documents by using latent semantic analysis and SOM*, E. Oja and S. Kaski editors, Kohonen Maps, 1999.
23. D. Achlioptas, F. McSherry, and B. Scholkopf, "Sampling techniques for kernen methods," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 335–342, 2001.
24. I. Rigoutsos and A. Califano, "Flash: A fast look-up algorithm for string homology," in *International Conference on Intelligent Systems for Molecular Biology*, pp. 221–227, 1993.
25. J. Buhler and M. Tompa, "Finding motifs using random projection," *Journal of Computational Biology* **9, no. 2**, pp. 225–242, 2002.
26. S. Chandrasekaran, B. S.Manjunath, Y.F.Wang, J.Winkeler and H.Zhang, "An eigenspace update algorithm for image analysis," *Graphical Models and Image Processing* **59, no. 5**, pp. 321–332, 1997.
27. S. Kaski, "Dimensionality reduction by random mapping: Fast similarity computation for clustering," in *Proceedings of IEEE International Joint Conference on Neural Networks*, 1998.
28. ATT Laboratories Cambridge, "Orl face database, http://www.uk.research.att.com/facedatabase.html," tech. rep.
29. P. Peer, "Cvl face database, http://www.lrv.fri.uni-lj.si/facedb.html," tech. rep.
30. A.M. Martinez and R. Benavente, "Cvc technical report 24," tech. rep., Ph.D. Thesis, University of Cambridge and ATT Laboratories Cambridge, 1998.
31. J. Kittler, M. Hatef, R.P.W. Duin and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20, no. 3**, pp. 83–87, 1998.

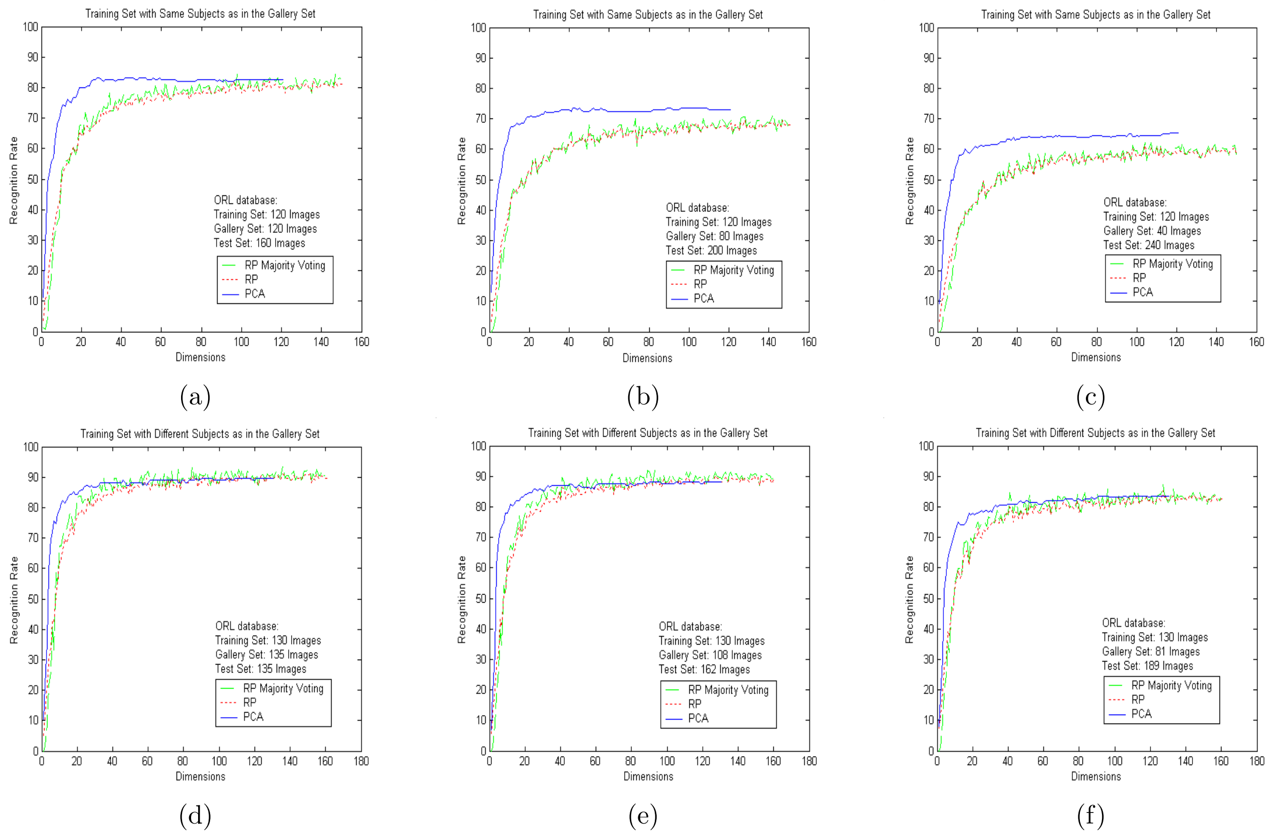


Figure 1. Experiments using the ORL database, closest match, and majority voting. (a)-(c) Same subjects in the training and gallery sets, (d)-(f) Different subjects in the training and gallery sets. The proportion of subjects in the gallery and test sets varies as shown. The blue line corresponds to PCA using closest match, the red line corresponds to RP using closest match, and the green line corresponds to RP using majority voting.

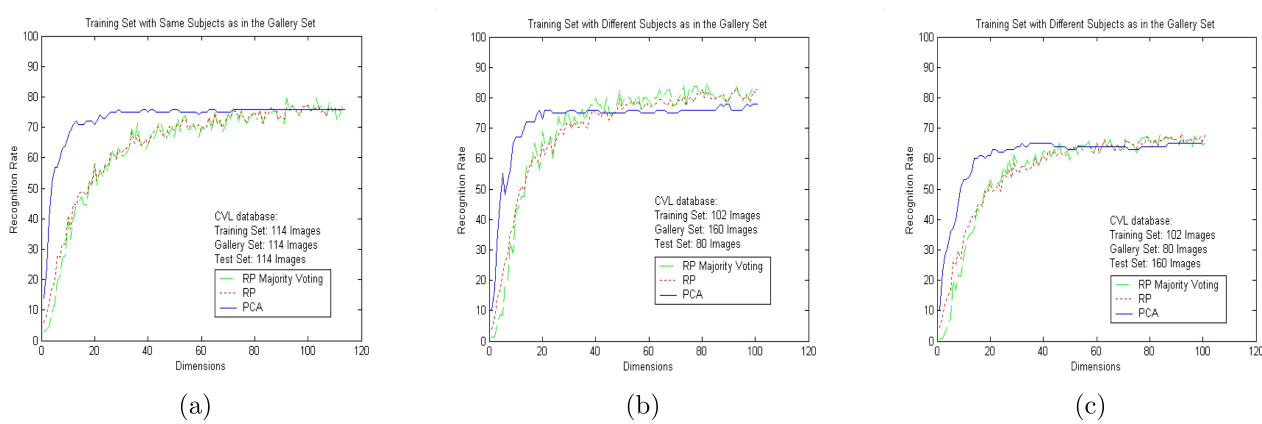
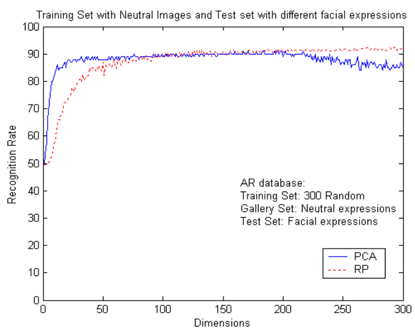
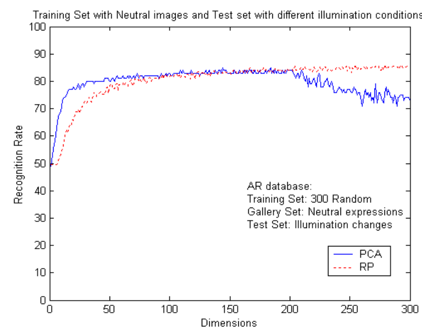


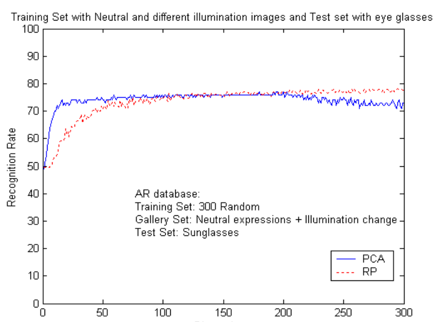
Figure 2. Experiments using the CVL database, closest match, and majority voting. (a) Same subjects in the training and gallery sets (b)-(c) Different subjects in training and gallery sets. The proportion of subjects in the gallery and test sets varies as shown. The blue line corresponds to PCA using closest match, the red line corresponds to RP using closest match, and the green line corresponds to RP using majority voting.



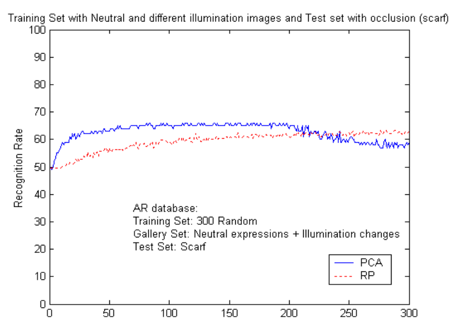
(a)



(b)



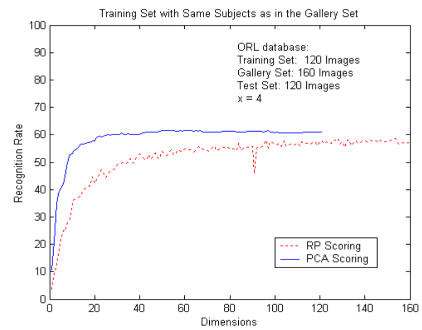
(c)



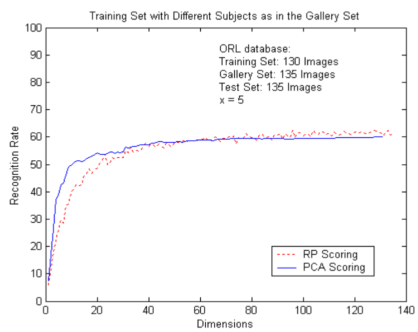
(d)

Figure 3. Experiments using the AR database and majority voting. The subjects in the training set were chosen randomly (i.e., not necessarily the same to those in the gallery set). The proportion of subjects in the gallery and test sets varies as shown. (a) neutral versus facial expressions, (b) neutral versus illumination, (c) neutral + illumination versus sunglasses, (d) neutral + illumination versus occlusion. The blue line corresponds to PCA using closest match, while the red line corresponds to RP using majority voting.

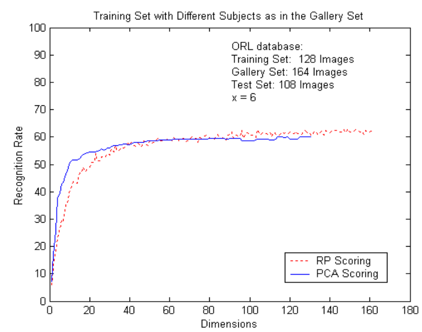
(a)



(b)



(c)



(d)

Figure 4. Experiments using the ORL database and scoring. (a)-(b) the subjects in the training and gallery sets were the same, (c)-(d) the subjects in the training and gallery sets were different. The blue line corresponds to PCA using scoring, while the red line corresponds to RP using scoring. The proportion of subjects in the gallery and test sets varies as shown.

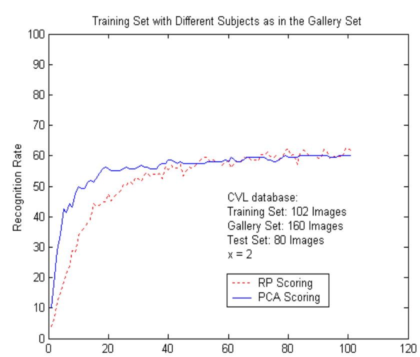
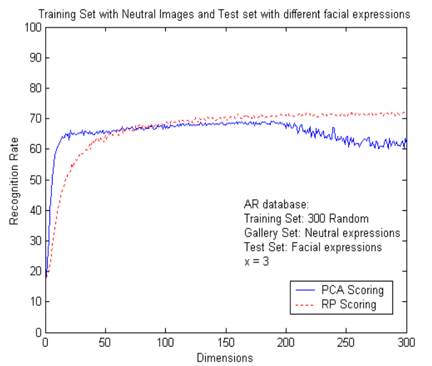
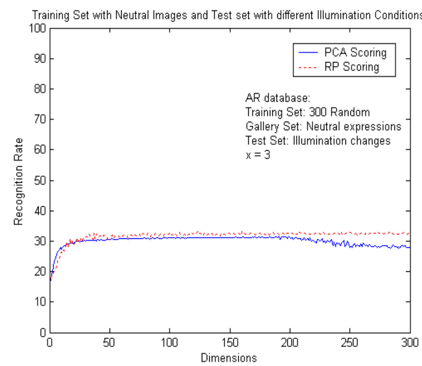


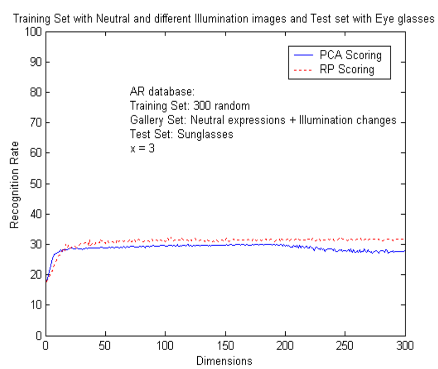
Figure 5. Experiments using the CVL database and scoring. The subjects in the training and gallery sets were different. The blue line corresponds to PCA using scoring, while the red line corresponds to RP using scoring. The proportion of subjects in the gallery and test sets varies as shown.



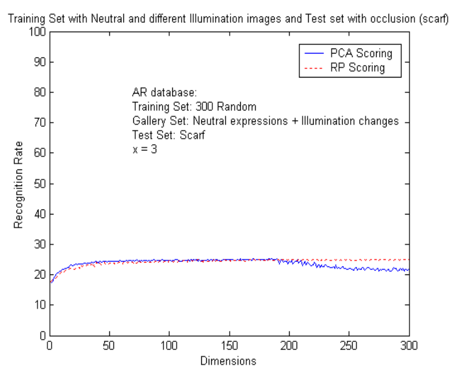
(a)



(b)



(c)



(d)

Figure 6. Experiments using the AR database and scoring. The subjects in the training set were chosen randomly (i.e., not necessarily the same to those in the gallery set). The proportion of subjects in the gallery and test sets varies as shown. The blue line corresponds to PCA using scoring, while the red line corresponds to RP using scoring.