HAL open science

Semi-supervised dimensionality reduction using pairwise equivalence constraints

Hakan Cevikalp, Jakob Verbeek, Frédéric Jurie, Alexander Klaser

To cite this version:

Hakan Cevikalp, Jakob Verbeek, Frédéric Jurie, Alexander Klaser. Semi-supervised dimensionality reduction using pairwise equivalence constraints. 3rd International Conference on Computer Vision Theory and Applications (VISAPP '08), Jan 2008, Funchal, Portugal. pp.489–496. inria-00321053v1

HAL Id: inria-00321053

https://inria.hal.science/inria-00321053v1

Submitted on 25 Jan 2011 (v1), last revised 11 Apr 2011 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SEMI-SUPERVISED DIMENSIONALITY REDUCTION USING PAIRWISE EQUIVALENCE CONSTRAINTS

Hakan Cevikalp, Jakob Verbeek, Frédéric Jurie, Alexander Kläser
Inria Rhone-Alpes, Montbonnot, France
cevikalp,jakob.verbeek,frederic.jurie,alexander.klaser@inrialpes.fr

Keywords: Constrained clustering, dimensionality reduction, image segmentation, metric learning, pairwise constraints, semi-supervised learning, spectral clustering.

Abstract: To deal with the problem of insufficient labeled data, usually side information – given in the form of pairwise equivalence constraints between points – is used to discover groups within data. However, existing methods using side information typically fail in cases with high-dimensional spaces. In this paper, we address the problem of learning from side information for high-dimensional data. To this end, we propose a semi-supervised dimensionality reduction scheme that incorporates pairwise equivalence constraints for finding a better embedding space, which improves the performance of subsequent clustering and classification phases. Our method builds on the assumption that points in a sufficiently small neighborhood tend to have the same label. Equivalence constraints are employed to modify the neighborhoods and to increase the separability of different classes. Experimental results on high-dimensional image data sets show that integrating side information into the dimensionality reduction improves the clustering and classification performance.

1 INTRODUCTION

Supervised learning techniques use training data with class labels being associated to the data samples. In many applications, there is a lack of labeled data since obtaining labels typically is a costly procedure as it often requires human effort. On the other hand, in some applications side information – given in the form of pairwise equivalence constraints between points – is available without or with little extra cost. For instance, faces extracted from successive video frames in roughly the same location can be assumed to represent the same person, whereas faces extracted in different locations in the same frame can be assumed to be from different persons. Side information may also come from human feedback, often at a substantially lower cost than explicit labeled data.

Existing learning methods that use side information to discover groups within data typically fall into one of two categories. The first category contains semi-supervised clustering methods which integrate equivalence constraints into the clustering process. This is accomplished by modifying the objective function such that constraints will be satisfied during the clustering. In (Wagstaff and Rogers, 2001; Basu et al., 2004), side information is integrated into the k-means clustering algorithm. Similarly, (Shental et al., 2003) and (Hertz et al., 2003) use equivalence constraints within the EM algorithm to estimate Gaussian mixture models. Methods in the second category revise the distance metric by warping the input space such that the constraints will be satisfied. They then perform clustering using the learned distance metric. In (Xing et al., 2003), a full rank Mahalanobis distance metric is learned using side information through convex programming. The metric is learned via an iterative procedure that involves projection and eigen-decomposition in each step. (Tsang and Kwok, 2003) formulate a full rank metric learning problem that uses side information in a quadratic optimization scheme. Using the kernel trick, the method is extended to the nonlinear case. In addition to these methods, a unified constrained-clustering and metric learning approach is proposed in (Bilenko et al., 2004).

Although the above approaches incorporate side information and yield satisfactory results for low-dimensional spaces, they typically fail for cases with high-dimensional spaces. This is due to the fact that most dimensions in high-dimensional spaces do not carry information about the class labels. Therefore they are likely to degrade the clustering performance. Furthermore, learning an effective full rank distance metric by using constraints in high-dimensional spaces is impracticable since (a) the number of parameters to be estimated is the square of the dimensionality, and (b) typically insufficient side information is available in order to obtain accu-

rate estimates. A typical solution to this problem is to reduce the dimensionality and to modify the distance metric in the reduced space, as in (Yan and Domeniconi, 2006). However, important information may be lost during a completely unsupervised dimension reduction (that does not use the side information) which may degrade the subsequent metric learning.

In this paper we propose a semi-supervised dimensionality reduction scheme which uses side information in the form of pairwise equivalence constraints to improve clustering and classification performance. The remainder of the paper is organized as follows. In Section 2 we present our approach, Section 3 describes the data sets and experiments, and we conclude in Section 4.

2 SEMI-SUPERVISED DIMENSIONALITY REDUCTION

2.1 Problem Setting

Let $X = [x_1 x_2 ... x_n]$ be a matrix whose columns contain d-dimensional samples. We are given a set of equivalence constraints in the form of similar and dissimilar pairs. Let S be the set of similar pairs

$$S = \left\{ (x_i, x_j) | x_i \text{ and } x_j \text{ belong to the same class} \right\}$$

and let D be the set of dissimilar pairs

$$D = \left\{ (x_i, x_j) | x_i \text{ and } x_j \text{ belong to different classes} \right\}.$$

Assuming consistency of the constraints, the constraint sets can be augmented using transitivity and entailment properties as in (Basu et al., 2004). Our goal is to find a lower-dimensional embedding space in which the equivalence constraints are satisfied.

Linear dimensionality reduction, that maps vectors x to lower dimensional vectors $y = A^\top x$, can be seen as learning a distance metric since the Euclidean distance between two points y_1 and y_2 in the reduced space can be written as

$$d(y_1, y_2) = \sqrt{(x_1 - x_2)^\top A A^\top (x_1 - x_2)}. \quad (1)$$

In this paper, our aim is to utilize the equivalence constraints for guiding the embedding process. To accomplish this goal, we use the Locality Preserving Projection (LPP) method (He and Niyogi, 2003) and modify its objective function to satisfy the equivalence constraints. Since our proposed method is based on LPP, we next recall the main idea of the method.

2.2 Locality Preserving Projections

The LPP method searches for an embedding space in which the similarity among the local neighborhoods is preserved. Firstly, an adjacency graph with n nodes is constructed. An edge between nodes i and j is created based on neighborhoods (e.g., using the k nearest neighbors). Then, each edge is weighted according to a similarity function. The weights W_{ij} lie in the range $[0,1]$ and take higher values for closer samples. The goal of LPP is to find the minimizer a^* of the loss function

$$E(a) = \frac{1}{2} \sum_{i,j} (y_i - y_j)^2 W_{ij}, \quad (2)$$

where a is the transformation vector, $y_i = a^\top x_i$ is one-dimensional representation of x_i, and W_{ij} is the weight between the vectors x_i and x_j. This loss function assigns a high penalty to mapping neighboring points x_i and x_j far apart. The loss function can be written in a more compact form as

$$E(a) = a^\top X (G - W) X^\top a = a^\top X L X^\top a, \quad (3)$$

where W is the matrix of weights and G is a diagonal matrix whose entries are the column (or row) sums of W. The matrix $L = G - W$ is called the Laplacian matrix. An additional constraint, $a^\top X G X^\top a = 1$, is included to normalize the projected data through G. The final transformation matrix A is constructed by the eigenvectors which are the minimum eigenvalue solutions to the generalized eigenvalue problem

$$X L X^\top a = \lambda X G X^\top a. \quad (4)$$

LPP has close ties with spectral clustering methods. Therefore, the LPP scheme can be defined through random walks similar to spectral clustering as shown in (Melia and Shi, 2001). A random walk on a graph is a stochastic process which randomly jumps from vertex to vertex. When the clustering is performed in the embedded space, the algorithm splits the data into clusters such that a random walk stays long within the same cluster and only rarely jumps between clusters. The transition probability of jumping in one step from vertex i to vertex j is proportional to the edge weight W_{ij}. When side information is available, the weights of adjacency matrix can be adjusted to reflect the equivalence constraints so as to find a better embedding. This is the main idea from which we develop our dimensionality reduction method in the following.

2.3 Integrating Equivalence Constraints

Similar to the dimensionality reduction methods that aim to preserve local structure, we assume that points

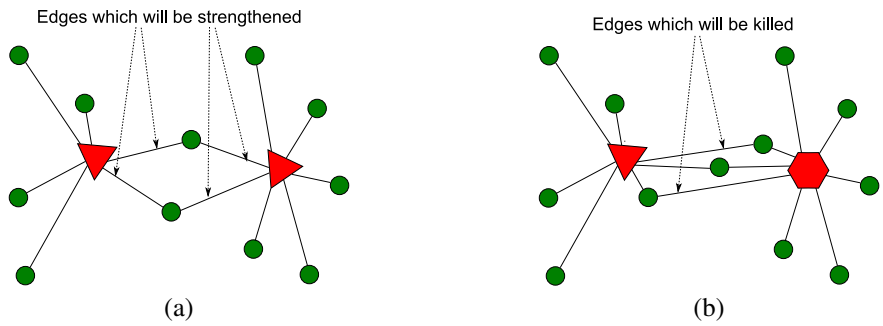

Figure 1: Propagating side information to the neighborhoods: (a) Similarity information is propagated by setting the edges of mutual neighbors to $+1$. (b) Dissimilarity information is propagated by killing the weakest edges.

in sufficiently small neighborhoods tend to have the same label. If constraints are chosen in a random fashion, then it is reasonable to expect that there will be equivalence constraints among non-neighboring samples. Such constraints are used to encourage mapping the involved points close to one another. This can be accomplished by setting the weight between non-neighboring points involved in equivalence constraints to a value larger than the original value of zero. Similarly, we can reset the edge weight between dissimilar points so that they will be pushed apart. Moreover, the constraint information may be propagated to the neighborhoods of the points involved in the constraints.

Given unlabeled data and equivalence constraints, our proposed method can be summarized as follows:

1. **Constructing Adjacency Graph.** We first construct a weighted graph with n nodes (one for each point) and a set of edges connecting neighboring points. Given a distance metric, we use k-nearest neighbors to determine the neighboring points. Node i and j are connected by an edge if i is among k-nearest neighbors of j, or vice versa. The neighborhood size k is set to a small number, e.g., $k=3$ or $k=5$. Each edge is then weighted by using the heat kernel (He and Niyogi, 2003) and a selected distance function, i.e.,

$$W_{ij} = \exp(-d(x_i,x_j)^2/t).$$

2. **Integrating Similarity Information.** If there is a similarity constraint between two points, say x_i and x_j, an edge is created and its weight is set to $+1$ (the highest similarity value). To propagate the similarity information to the neighbors, it is checked if x_i and x_j have common neighbors. If this is the case, the weight between each common neighbor x_k and x_i as well as x_k and x_j is set to $+1$, as illustrated in Fig. 1(a). This process strengthens the transition probability between similar points. If there is not any mutual neighbor, no action is taken to modify neighborhoods.

3. **Integrating Dissimilarity Information.** In the case of a dissimilarity constraint between two points, it is checked whether an edge exists for these points. Additionally it is checked whether those points share common neigbors. An edge, or common neighbors, between dissimilar points indicates that the involved dissimilar points are relatively close, which should be avoided. If an edge between dissimilar points exists, we set the edge weight to -1. In the case of common neighbors, we compare the similarities for each common neighbor to the dissimilar points. If one of those edges has a significantly lower weight, it is removed, as illustrated in Fig. 1(b). Otherwise no action is taken.

The objective for the above described method can be written as

$$E'(a) = \frac{1}{2}\Big(\sum_{i,j}(y_i-y_j)^2\widetilde{W}_{ij} + \sum_{i,j\in S}(y_i-y_j)^2 - \sum_{i,j\in D'}(y_i-y_j)^2\Big). \quad (5)$$

where $\widetilde{W}_{ij}$ represents the updated version of the weights, and D' denotes the set of dissimilar points that were originally neighbors or had common neighbors. Our objective function has three terms. The first term targets preserving the modified local structure of the data. The second term aims at pulling the similar points closer, whereas the last term encourages pushing apart dissimilar points that are nearby in the graph. The final transformation matrix A that minimizes this cost function typically includes the difference directions between dissimilar points coming from D' since the dissimilar points can be pushed apart by using these directions. Including those difference directions is important since they participate in shaping the inter-class decision boundaries.

We can rewrite the cost function as

$$E'(a) = a^\top X(G_S + G_N + G_{D'} - W_S - W_N - W_{D'})X^\top a$$
$$= a^\top X(G' - W')X^\top a,$$

where W_S is the sparse weight matrix corresponding to pairs in S with edge weights $+1$, $W_{D'}$ is the sparse weight matrix of pairs in D' with edges weights -1, finally W_N is the adapted weight matrix between original neighbors with edge weights $\widetilde{W}_{ij}$. The matrices G_S, G_N, and $G_{D'}$ are diagonal matrices containing the row sums of the corresponding W matrix.

As in LPP, we introduce the constraint $a^\top XG'X^\top a = 1$ to fix the scale of a. The final transformation matrix A is constructed by the minimum-eigenvalue eigenvectors of the generalized eigenvector equation

$$X(G' - W')X^\top a = \lambda XG'X^\top a. \quad (6)$$

We coin our method *Constrained Locality Preserving Projections (CLPP)* since it allows one to use pairwise equivalence constraints in the LPP method.

2.4 Extension to Non-linear Projections

Our method can produce non-linear projections using the kernel trick. Suppose that the samples in the original input space $\mathbb{R}^d$ are mapped to a higher-dimensional feature space $\mathcal{F}$ using a nonlinear mapping function $\Phi : \mathbb{R}^d \to \mathcal{F}$. Let $\Phi(X) = [\Phi(x_1)\Phi(x_2)\ldots\Phi(x_n)]$ denote the matrix whose columns are the mapped samples in $\mathcal{F}$. We then search for a linear projection in $\mathcal{F}$, which leads to the eigenvalue equation

$$\Phi(X)(G' - W')\Phi(X)^\top a = \lambda\Phi(X)G'\Phi(X)^\top a. \quad (7)$$

Since the eigenvectors are linear combinations of the mapped samples, there exist coefficients α_i ($i = 1,\ldots,n$), such that

$$a = \sum_{i=1}^{n} \alpha_i \Phi(x_i) = \Phi(X)\alpha. \quad (8)$$

The dot products in the feature space $\mathcal{F}$ is computed through a Mercer kernel $k(\cdot,\cdot)$. Let $K = \Phi(X)^\top\Phi(X) = (k(x_i,x_j))_{i,j}$ denote the kernel matrix of the data samples. Multiplying Eq. (7) on the left with $\Phi(X)^\top$, the eigenvector equation is converted to

$$K(G' - W')K\alpha = \lambda KG'K. \quad (9)$$

Let α be one of the minimum eigenvalue solutions to the above equation, then the data projections in $\mathcal{F}$ are computed as $y = K\alpha$ where the i-th element of y is the one-dimensional representation of x_i. If rather than using $y = K\alpha$ we allow for general data representations y, the solutions are given by

$$(G' - W')y = \lambda G'y, \quad (10)$$

which may be interpreted as the Laplacian Eigenmap (Belkin and Niyogi, 2001) solution of the modified graph.

3 EXPERIMENTAL EVALUATION

3.1 Methodology and Data Sets

To assess the performance of our method, we have performed experiments on two image databases – ETH-80 (Leibe and Schiele, 2003) and Birds (Lazebnik et al., 2005) – to discover object groups. Several example images from the databases are shown in Fig. 2. We used only four categories from the ETH-80: *Apple, Car, Cow,* and *Cup*. Each category contains images of 10 to 14 objects under different viewpoints, against a flat blue background. The Birds database contains six categories where each category includes 100 images. It is a challenging database including images with large intra-class, scale, and viewpoint variability. Furthermore, birds appear against highly cluttered backgrounds.

We used a 'bag of features' image representation. In this approach, patches are sampled from the image at many different positions and scales, either densely, randomly or based on the output of some kind of salient region detector. In our case we select patches following a dense grid. Then each patch is represented by a 128-dimensional SIFT descriptor (Lowe, 2004). Following this process, all descriptors extracted from images are quantized in a discrete set of so-called 'visual keywords' forming a vocabulary. To build image representation, each extracted descriptor is compared to the visual keywords and associated to the closest keyword. Based on these assignments, we build histograms which are used as image feature vectors. The size of the histograms is 500 and 2000 for the ETH and Birds datasets, respectively. It has been shown that the Chi-square distance is well-suited for measuring the similarity among the histograms (Cevikalp et al., 2007). Therefore we utilize Chi-square distances in the Heat kernel function when building the initial weight matrix W.

To show the efficacy of using equivalence constraints for discovering the hidden groups within data, we apply k-means clustering in the embedded space and use pairwise F-measure to evaluate the clustering results based on the underlying classes. The pairwise

Figure 2: ETH-80 (top row) and Birds (the second and third rows) datasets: 2 illustrative images per category.

F-measure is the harmonic mean of the pairwise precision and recall measures which are widely used in information retrieval. We compute precision and recall over pairs of images and consider for the pairs whether they are assigned to the same cluster by k-means and whether they contain the same object category. Let A denote the set of image pairs assigned to the same k-means cluster, and let B denote the set of image pairs that contain the same object category. With $|A|$ denoting the cardinality of A (and similar for other sets), the measures are defined as:

$$\text{Precision} = \frac{|A \cap B|}{|A|}, \qquad \text{Recall} = \frac{|A \cap B|}{|B|},$$

$$\text{F-measure} = \frac{2 \times Precision \times Recall}{Precision + Recall}.$$

Performance evaluations were obtained using cross-validation; 5-fold for the ETH data set and 4-fold the Birds data set. The clustering algorithm was run on the whole data set, but the F-measure was computed for the whole data set and the held-out test set separately.

To demonstrate the effect of using different number of equivalence constraints, beginning without constraints we gradually increase the number of similar and dissimilar pairs. In all experiments constraints are uniformly random selected from all possible constraints induced by the true data labels of the training data.

As mentioned in the introduction, we cannot apply full rank distance metric learning techniques in these high-dimensional spaces. To compare our method CLPP to other distance metric learning techniques, we first applied dimensionality reduction methods, Principal Component Analysis (PCA) and LPP, to the high-dimensional data and learned a distance metric in the reduced space. To learn the distance metric, we applied the methods proposed in (Tsang and Kwok, 2003) and (Xing et al., 2003). The former yields better results, therefore we only report results for this method.

3.2 Experimental Results

F-measure scores are shown in Fig. 3. As can be seen in the results, adding constraints improves the clustering performance. For the ETH data set, our proposed method and the LPP followed by full rank metric learning technique yield similar results. On the other hand, the proposed method significantly outperforms the full rank metric learning approach for the Birds dataset. It is because most of the discriminatory information is lost during the unsupervised dimensionality reduction stage. Therefore the metric learning stage improves the clustering performance up to some degree in the reduced space and then saturates even if additional constraints are used. On the contrary, utilizing constraints in the proposed dimensionality reduction scheme achieves better results, and adding new constraints continues to improve the clustering performance. In Fig. 5, we plot the affinity matrices in the original sample space and the embedded space. As can be seen in the figure, adding constraints increases the class separability, which explains the increase in clustering performance.

We also conducted experiments to show how the proposed method improves the distance metric and classification performance in the projected space. To this end, from the Birds dataset, we randomly selected 10000 sample pairs which are not used as similar and dissimilar pairs. Then, we converted the problem to a binary classification problem treating the pairs coming from same classes as positive samples and pairs coming from different classes as negative samples.

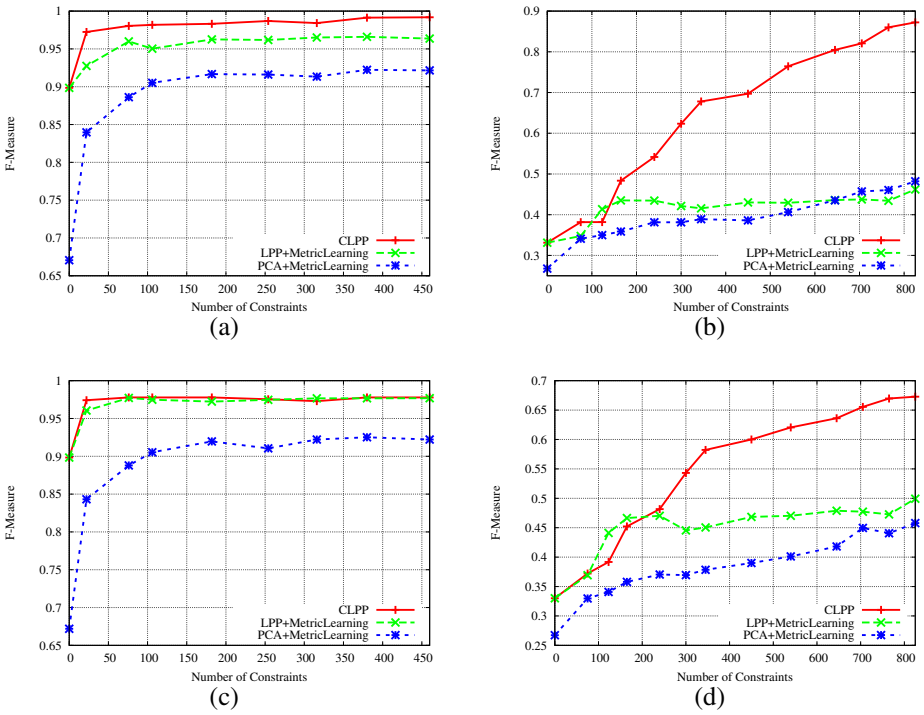

Figure 3: F-measure as a function of number of constraints for (a) overall ETH data, (b) overall Birds data, (c) ETH held-out test data, and (d) Birds held-out test data.

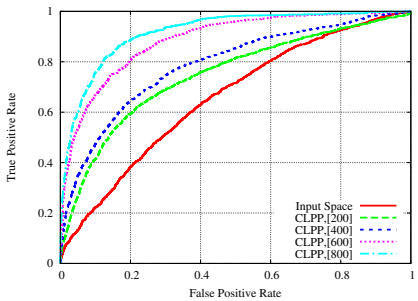

Figure 4: ROC curves for the Birds Database.

We then computed the Euclidean distances in the projected CLPP space. Based on the these distances we created Receiver Operating Characteristic (ROC) curves. This procedure is also repeated in the original input space by using the Chi-Square distances. Curves are plotted in Fig. 4 for different number of constraints given in square brackets. From the ROC curves, we see that as the number of constraints is increased, the accuracy of the classification procedure improves indicating that our embedding procedure improves the original distance metric.

3.3 Image Segmentation Applications

The proposed CLPP method can also be applied for clustering low-dimensional data samples by using the kernel trick. To test the efficacy of the kernel method we applied it in image segmentation task. We have chosen five images from the Berkeley Segmentation dataset[1]. Centered at every pixel in each image we extracted a 20×20 pixel image patch for which we computed the robust hue descriptor of (van de Weijer and Schmid, 2006). This process yields a 36-dimensional feature vector which is a histogram over hue values observed in the patch, where each observed hue value is weighted by its saturation. The Heat kernel function using Euclidean distance is used as kernel. We set the number of clusters to two, one cluster for the background and another for the object of interest.

The pairwise equivalence constraints are chosen from the samples corresponding to pixels shown with magenta and cyan in the second row of Fig. 6. We first segmented the original images (top row) without using constraints (result in the third row) and then we used constraints for segmentation (result in bot-

[1] Available at `http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/`

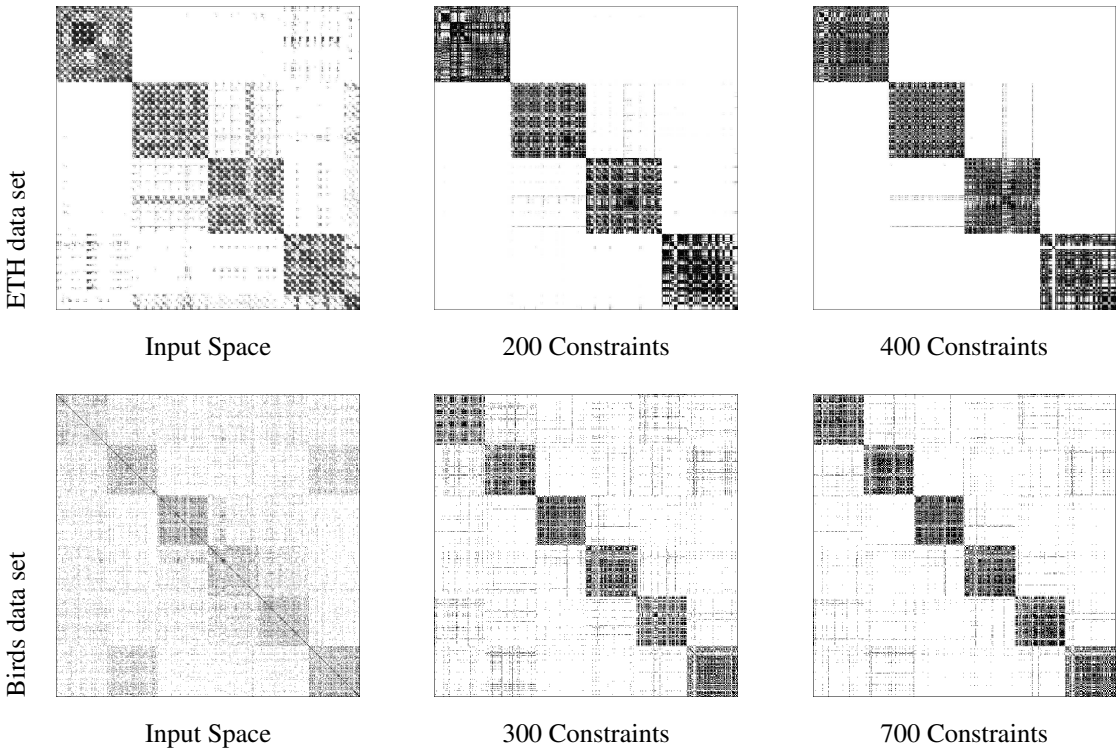

Figure 5: Visualization of affinity matrices obtained from the ETH dataset (first row) and Birds dataset (second row).

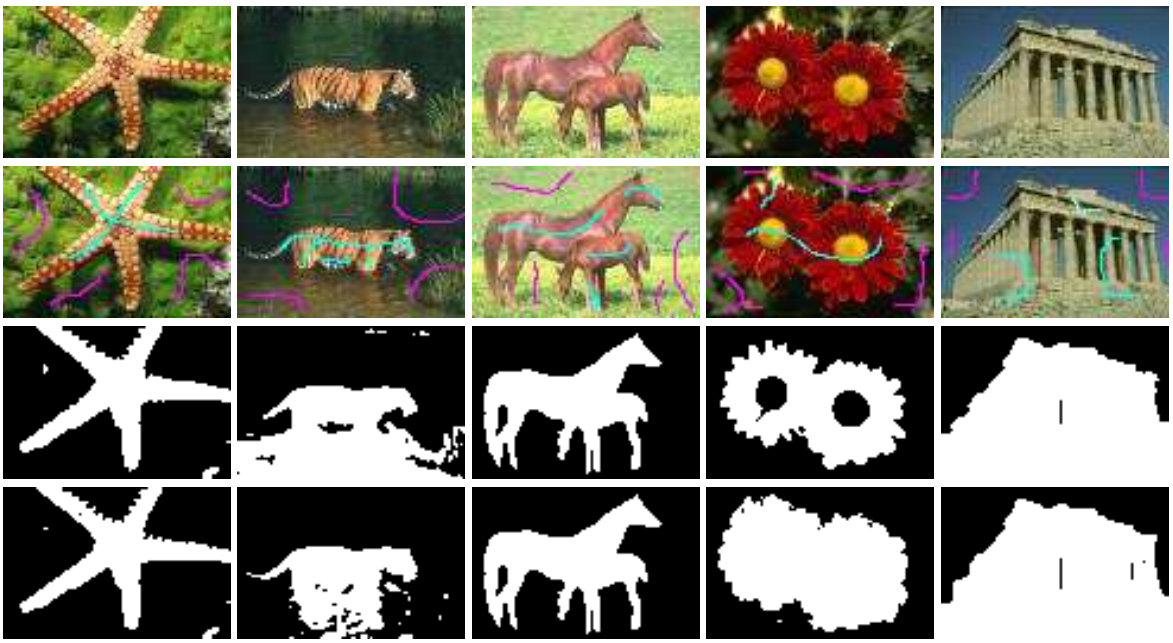

Figure 6: Original images (top row), pixels used for equivalence constraints (second row), segmentation results without constraints (third row), and segmentation results using constraints (bottom row). Figure is best viewed in color.

tom row). As can be seen in the figure, simple user added (dis)similarity constraints can significantly improve the segmentations. Consider for instance the flower image, there are three well separated color components in the image: the green background, the red leaves, and the yellow flower center. There are thus three reasonable segmentations –separating each one of the components from the other two– and it is a-priori not clear which is desired by a user. However once a small set of (dis)similarity constraints are added, the segmentation desired by the user is easily identified.

4 CONCLUSION

In this paper we developed a semi-supervised dimensionality reduction method which uses pairwise equivalence constraints to discover the groups in high-dimensional data. To this end, we modified LPP scheme such that its objective function takes into account the equivalence constraints. Like LPP, our algorithm first finds neighboring points to create a weighted neighborhood graph. Then, the constraints are used to modify the neighborhood relations and weight matrix to reflect this weak form of supervision. The optimal projection matrix according to our cost function is then identified by solving for the smallest eigenvalue solutions of an $n \times n$ eigenvector problem, where n is the number of data points. Experimental results show that our semi-supervised dimensionality reduction method increases performance of subsequent clustering and classification algorithms. Moreover, it yields better results than methods applying unsupervised dimensionality reduction followed by full-rank metric learning.

In some applications, small subsets of data points with same class labels, so-called 'chunklets', occur naturally, e.g., for face recognition in video. In future work, we will explore distance metrics between chunklets as well as chunklets and points, rather than between individual data points. Since these metrics operate on richer data structures, we expect them to significantly improve clustering and classification results.

REFERENCES

Basu, S., Banerjee, A., and Mooney, R. J. (2004). Active semi-supervision for pairwise constrained clustering. In *the SIAM International Conference on Data Mining*.

Belkin, M. and Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*.

Bilenko, M., Basu, S., and Mooney, R. J. (2004). Integrating constraints and metric learning in semi-supervised clustering. In *the 21st International Conference on Machine Learning*.

Cevikalp, H., Larlus, D., Neamtu, M., Triggs, B., and Jurie, F. (2007). Manifold based local classifiers: Linear and nonlinear approaches. In *Pattern Recognition in review*.

He, X. and Niyogi, P. (2003). Locality preserving directions. In *Advances in Neural Information Processing Systems*.

Hertz, T., Shental, N., Bar-Hillel, A., and Weinshall, D. (2003). Enhancing image and video retrieval: Learning via equivalence constraints. In *the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03)*.

Lazebnik, S., Schmid, C., and Ponce, J. (2005). A maximum entropy framework for part-based texture and objcect recognition. In *International Conference on Computer Vision (ICCV)*.

Leibe, B. and Schiele, B. (2003). Interleaved object categorization and segmentation. In *British Machine Vision Conference (BMVC)*.

Lowe, D. (2004). Distinctive image features from scale - invariant keypoints. In *International Journal of Computer Vision*, volume 60, pages 91–110.

Melia, M. and Shi, J. (2001). A random walks view of spectral segmentation. In *the 8th International Workshop on Artificial Intelligence and Statistics*.

Shental, N., Bar-Hillel, A., Hertz, T., and Weisnhall, D. (2003). Computing gaussian mixture models with em using equivalance constraints. In *Advances in Neural Information Processing Systems (NIPS)*.

Tsang, I. W. and Kwok, J. T. (2003). Distance metric learning with kernels. In *the International Conference on Artificial Neural Networks*.

van de Weijer, J. and Schmid, C. (2006). Coloring local feature extraction. In *European Conference on Computer Vision (ECCV)*.

Wagstaff, K. and Rogers, S. (2001). Constrained k-means clustering with background knowledge. In *the 18th International Conference on Machine Learning*.

Xing, E. P., Ng, A. Y., Jordan, M. I., and Russell, S. (2003). Distance metric learning with application to clustering with side-information. In *Advances in Neural Information Processing Systems*.

Yan, B. and Domeniconi, C. (2006). Subspace metric ensembles for semi-supervised clustering of high dimensional data. In *the 17th European Conference on Machine Learning*.