Dimensionality reduction with the k-associated optimal graph applied to image classification

Thiago H. Cupertino, Murillo G. Carneiro, Liang Zhao
Institute of Mathematics and Computer Science, ICMC
University of São Paulo, USP
São Carlos, Brazil 13566-590
Email: {thiagohc, carneiro, zhao}@icmc.usp.br

***Abstract*—In this paper, we aim to study the usage of different network formation methods into a graph embedding framework to perform supervised dimensionality reduction. Images are often high-dimensional patterns, and dimensionality reduction can enhance processing and also increase classification accuracy. Specifically, our technique maps images into networks and constructs two network adjacency matrices to convey information about intra-class components and inter-class penalty connections. Both matrices are inserted into an optimization framework in order to achieve a projection vector that is used to project high-dimension data samples into a low-dimensional space. One advantage of the technique is that no parameter is required, that is, there is no need to select a model for the input data. Applications on handwritten digits recognition are performed, and the proposed technique is compared to some classical network formation methods. Numerical results show the approach is promising.**

***Keywords*- Dimensionality reduction; network-based learning; supervised learning; marginal fisher criterion.**

I. Introduction

Many applications in data mining, machine learning and pattern recognition face problems when computing similarities among data samples. When data lies in a high-dimensional space, these problems are often due to the "curse of dimensionality" [1]. In this situation, similarity measures among data suffers from distortions, that is, when the dimensionality increases, the volume of the space increases so fast that the available data samples becomes sparse. Specially, this situation is often found when dealing with images, which possess a large dimensional feature space, that is, images are high-dimensional patterns. One way to alleviate this problem is by performing dimensionality reduction, which aims at reducing the dimension of the input data in order to achieve a small set of features that keeps the most important original relationships among data samples. This reduction can enhance image processing and also increase the classification accuracy [2], [3], [4], [5].

Techniques for dimensionality reduction often lie in the unsupervised or in the supervised learning. A classical example of unsupervised technique is the Principal Component Analysis (PCA) [6]. PCA is an orthogonal transformation that represent data by using the so called principal components. Usually, a small number of principal components is sufficient to account for most of the structure in the data. It maximizes the mutual information between the original high-dimensional Gaussian distributed measurements and the projected low-dimensional measurements. As an unsupervised technique, PCA does not use the class label information of the input data. In the supervised setting, data instances are marked with label information that guides the formation of the low-dimensional space. The labels often take discrete class values, indicating which data points have to be grouped together (same class) or set far apart from the other (different classes) in the embedded space. In the group of supervised techniques, Linear Discriminant Analysis (LDA) [7] plays an important role. As a supervised technique, it uses the class label information of the input data samples. LDA finds a projection matrix that maximizes the trace of the between-class scatter matrix and minimizes the trace of the within-class scatter matrix in the projected subspace simultaneously.

Supervised dimensionality reduction can also be performed by using a graph embedding framework [8]. Graphs are powerful tools to represent data relationships and have been applied to a variety of learning tasks [9], [10], [11], [12], [13]. The purpose of graph embedding is to represent each vertex (data sample) of a network as a low-dimensional vector that preserves similarities between the vertex pairs, where similarity is measured by a graph similarity matrix that characterizes certain statistical or geometric properties of the data set. The usage of graph embedding for dimensionality reduction can overcome some limitations of the LDA technique such as the number of available projection directions lower than the number of classes, and the assumption that data is approximately Gaussian distributed [8].

In this paper, we study the usage of the recently proposed k-Associated Optimal Graph (KAOG) [14] into the graph embedding framework for dimensionality reduction. The KAOG is a network construction technique which relies on two concepts: a purity measure, which uses the graph representation to measure mixing levels of the original data samples regarding their classes given a k-neighborhood; and the k-associated graph, which can be considered as an improved adaptive k-Nearest Neighbor (k-NN) graph. The network construction process consists of building the k-associated optimal graph, that represents the data set as a sparse network in which components carry local information about the underlying data distribution [14]. Furthermore, we propose a modification of the KAOG network formation to construct a penalty graph, which is required for the graph embedding framework. The penalty graph conveys information about which data samples (class components) should not be close together (different classes) in the reduced feature space. The proposed technique is compared to two classical network construction methods: k-NN and ϵ-radius.

This paper is organized as follows. Section II introduces the problem setting of dimensionality reduction. Section III describes the network formation methods to construct the scatter-matrices to be used into the graph embedding framework. Section IV shows the experimental results and section V concludes the paper.

II. Dimensionality reduction problem setting

In this paper, we consider that it is given a training data set $\mathcal{X}^{(l)} = \{\mathbf{x}_i^{(l)},\ i = 1, \ldots, n\}$, containing labeled images, and a test data set $\mathcal{X}^{(u)} = \{\mathbf{x}_i^{(u)},\ i = 1, \ldots, m\}$, containing unlabeled images. Each image is described by q attributes, that is, a vector $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{iq}]^T$, and belongs to a single class $c \in \{1, \ldots, C\}$, where C is the number of classes. The goal of the proposed technique is to perform dimensionality reduction by using the information provided by the labeled data set $\mathcal{X}^{(l)}$ in order to improve classification accuracy or, at least, to speed up the classification process of the unlabeled data set $\mathcal{X}^{(u)}$ without decreasing the accuracy, given that a small number q' of projected attributes is used ($q' < q$).

Usually, the image feature dimension q can be very high, and transforming the data from the original high-dimensional space to a low-dimensional space can alleviate the curse of dimensionality [1]. To accomplish that, a technique should find a mapping function F that transforms $\mathbf{x}$ into the desired low-dimensional representation $\mathbf{y}$, so that $\mathbf{y} = F(\mathbf{x})$ ($\mathbf{y} \in \mathbb{R}^{q'}$). By using an underlying network to find such function F, the dimensionality reduction process can be viewed as a graph-preserving criterion of the following form [8]:

$$Y^* = arg\,min \sum_{i \neq j} ||\mathbf{y}_i - \mathbf{y}_j||^2 W_{ij} = arg\,min\, Y^T L Y, \quad (1)$$

constrained to $Y^T B Y = \mathbf{d}$. In this formulation, $\mathbf{d}$ is a constant vector, W_{ij} is the adjacency matrix of the network, B is the constraint matrix and L is the Laplacian matrix. The Laplacian matrix can be found via the following operation:

$$L = D - W,\ Dii = \sum_{i \neq j} W_{ij}, \forall i.$$

The constraint matrix B can be viewed as the adjacency matrix of a penalty network W^P, so that $B = L^P = D^P - W^P$. The penalty network conveys information about which vertices should not be linked together, that is, which instances should be far apart after the dimensionality reduction process. The similarity preservation property from the graph-preserving criterion has a two-fold explanation. For larger similarity between samples $\mathbf{x}_i$ and $\mathbf{x}_j$, the distance between $\mathbf{y}_i$ and $\mathbf{y}_j$ should be smaller to minimize the objective function. Likewise, smaller similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$ should lead to larger distances between $\mathbf{y}_i$ and $\mathbf{y}_j$ for minimization [8].

In this paper, we assume that the low-dimensional attribute space can be found by using a linear projection such as $Y = X^T \mathbf{w}$, in which $\mathbf{w}$ is the projection vector. The objective function in Eq. 1 becomes:

$$\begin{aligned} \mathbf{w}^* = & \quad arg\,min \sum_{i \neq j} ||\mathbf{w}^T x_i - \mathbf{w}^T x_j||^2 W_{ij} \\ = & \quad arg\,min\, \mathbf{w}^T X L X^T \mathbf{w}, \end{aligned} \quad (2)$$

constrained to $\mathbf{w}^T X L X^T \mathbf{w} = d$. By using the Marginal Fisher Criterion [8] and the penalty network constraint, Eq. 2 becomes:

$$\mathbf{w}^* = arg\,min_{\mathbf{w}} \frac{\mathbf{w}^T X L X^T \mathbf{w}}{\mathbf{w}^T X L^P X^T \mathbf{w}}, \quad (3)$$

which can be solved by the generalized eigenvalue problem by using the equation $X L X^T \mathbf{w} = \lambda X L^P X^T \mathbf{w}$.

III. Network formation techniques

The construction of the underlying networks is an elementary step of the proposed dimensionality reduction technique. In the literature, there are a few techniques related to network construction. The most used techniques are ϵ-radius and k-Nearest Neighbors (k-NN) [15]. However, both require parameter selection. On the other hand, a recently proposed technique, KAOG, provides a network that is constructed from a purity measure, without requiring the usage of parameter selection [14], [10].

In the next subsections, we provide the concepts related to each network formation technique and propose the adaptations that we develop to employ them in a new dimensionality reduction technique. We also illustrate the constructed networks by using each algorithm for the artificial data set showed in Fig. 1.

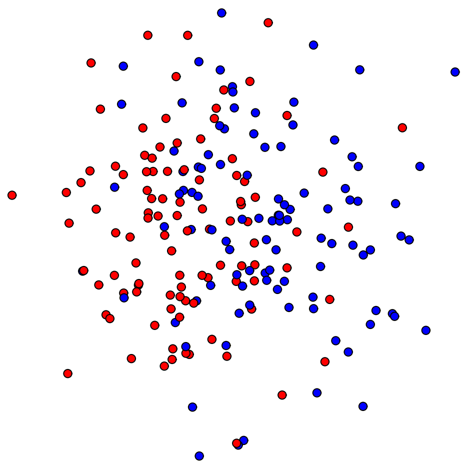

Fig. 1: Artificial data set composed of two mixed gaussians.

A. ϵ-Radius Network

In data classification, the ϵ-radius technique creates a link between two vertices i and j if two conditions are satisfied: i and j are within a distance ϵ and they belong to the same class:

$$E = E \cup \{e_{i,j} \mid d_{i,j} \leq \epsilon \; \& \; c_i = c_j\} \qquad (4)$$

The ϵ-radius technique provides a network with higher density when compared to other graph formation techniques. An example of the ϵ-radius network is illustrated in the Fig. 2, which presents the network formation using the data set showed in Fig. 1. Note that there are a large number of links among the vertices.

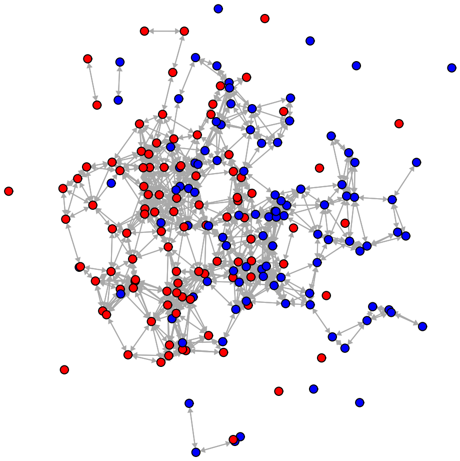

Fig. 2: Network constructed from ϵ-radius technique on a data set of two mixed gaussians. ϵ = 30% of the average distance among all vertices.

As explained in section II, our technique requires the construction of two matrices: the adjacency matrix and the penalty matrix. The adjacency matrix (E) is obtained directly from (4). Alg. 1 presents a simple way to obtain the penalty matrix B. The algorithm creates a link between i and j in B if the vertices are within a distance ϵ and belong two different classes. In this case, a link means that these vertices should be far apart after the dimensionality reduction process.

Algorithm 1 ϵ-radius algorithm

```
Require: $\epsilon$ and a data set $X$
$E, B \Leftarrow \emptyset$
for all $i, j \in X$ do
    if $d_{i,j} \leq \epsilon$ & $c_i = c_j$ then
        $E \Leftarrow E \cup e_{i,j}$
    else if $c_i \neq c_j$ then
        $B \Leftarrow B \cup e_{i,j}$
    end if
end for
return $E$ and $B$
```

B. k-NN Network

The k-NN network construction creates a link between vertices i and j if two conditions are satisfied: j is one of the k-nearest neighbors of i and the classes of i and j are the same, as showed by 5:

$$E = E \cup \{e_{i,j} \mid j \in K\text{-NN}(i) \; \& \; c_i = c_j\}. \qquad (5)$$

Unlike ϵ-radius network formation, k-NN is able to represent sparse regions of the network. Fig. 3 illustrates the application of the k-NN technique on the data set showed in Fig. 1. Note that vertices in sparse regions, which do not link using ϵ-radius technique (Fig. 2), are able to make connections using k-NN formation graph.

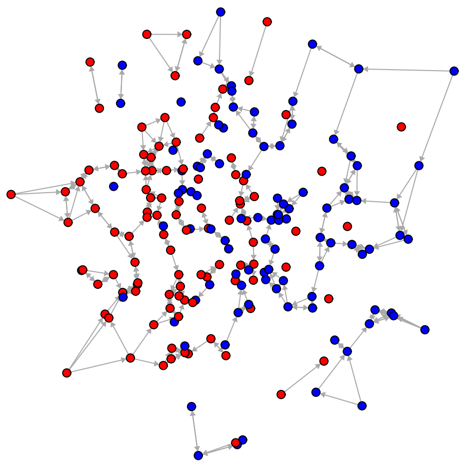

Fig. 3: Network constructed from k-NN technique on a data set of two mixed gaussians. k = 3.

We propose a simple way to obtain the penalty matrix B for the k-NN network as follows. There is a link from i to j in B only if the j is one of the k-nearest neighbors of i and their classes are distinct. In consequence, the adjacency matrix E is obtained from (5). Alg. 2 presents the steps to obtain E and B.

Algorithm 2 k-NN algorithm

```
Require: $K$ and a data set $X$
$E, B \Leftarrow \emptyset$
for all $i, j \in X$ do
    if $j \in K\text{-NN}(i)$ & $c_i = c_j$ then
        $E \Leftarrow E \cup e_{i,j}$
    else if $c_i \neq c_j$ then
        $B \Leftarrow B \cup e_{i,j}$
    end if
end for
return $E$ and $B$
```

C. KAOG Network

Differently from usual graph formation techniques, the KAOG technique constructs a network guided by a measure

Algorithm 3 k-Associated Optimal Graph

```
Require: data set X
$k \Leftarrow 1$
$G^{(op)} \Leftarrow k\text{-}associated\ graph(k, X)$ (Algorithm 4)
repeat
    $lastAvgDegree \Leftarrow D^{(k)}$
    $k \Leftarrow k + 1$
    $G^{(k)} \Leftarrow k\text{-}associated\ graph(k, X)$
    for all $C_\beta^{(k)} \subset G^{(k)}$ do
        if $\Phi_\beta^{(k)} \geq \Phi_\alpha^{(op)}$ for all $C_\alpha^{(op)} \subseteq C_\beta^{(k)}$ then
            $G^{(op)} \Leftarrow G^{(op)} - \cup_{C_\alpha^{(op)} \subseteq C_\beta^{(k)}} C_\alpha^{(op)}$
            $G^{(op)} \Leftarrow G^{(op)} \cup \{C_\beta^{(k)}\}$
        end if
    end for
until $D^{(k)} - lastAvgDegree < D^{(k)}/k$
return $G^{(op)}$
```

named purity. This measure expresses the level of mixture of a component in relation to other components of distinct classes and it is given by:

$$\Phi_\alpha = \frac{D_\alpha}{2K_\alpha}, \tag{6}$$

where D_α and k_α denote, respectively, the average degree and the k value associated to the component α. In this way, KAOG uses the purity measure to construct and optimize each component of the network.

Algorithm 3 shows step by step the construction of KAOG networks. Note that no parameter is needed by the algorithm. After the initial setting, a loop starts to merge the subsequent k-associated graphs by increasing k, while improving the purity of the network encountered so far, until the optimal network measured by the purity degree is reached. Basically, the k-associated graph (KAG) algorithm links a vertex (image) i to all its k-nearest neighbors that belong to the same class of i (a set denoted by $\Lambda_{i,k}$). More details about the algorithm are presented in [14].

Furthermore, we develop a fast way to obtain the penalty matrix B for the KAOG network as follows. There is a link between i and j in B if j is one of the k nearest neighbors of i, and j belong to a different class of i. Alg. 4 shows step by step how the links of the adjacency matrix E and the constraint matrix are done in the k-associated graph. It is worth noting that the constraint matrix is optimized by the purity measure too.

Figure 4 illustrates the construction of the KAOG network in a data set composed by two mixed gaussians. The resulted network is distinct from the ϵ-radius and k-NN techniques. The main advantages on these algorithms is that KAOG network is obtained without any parameter. In addition, its vertices are linked according to the maximization of the purity measure. This provides an optimized network and a robust mechanism to avoid noisy and outliers [14].

Algorithm 4 k-Associated Graph

```
Require: k and a data set X
$E, B \Leftarrow \emptyset$
for all $i \in V$ do
    if $j \in \Lambda_{i,k}$ & $c_i = c_j$ then
        $E \Leftarrow E \cup e_{i,j}$
    else if $c_i \neq c_j$ then
        $B \Leftarrow B \cup e_{i,j}$
    end if
end for
$C \Leftarrow findComponents(E)$
for all $\alpha \in C$ do
    $\Phi_\alpha \Leftarrow$ Eq. (6)
    $G^{(k)} \Leftarrow G^{(k)} \cup \{(\alpha(V', E', B'); \Phi_\alpha)\}$
end for
return k-associated graph $G^{(k)}$
```

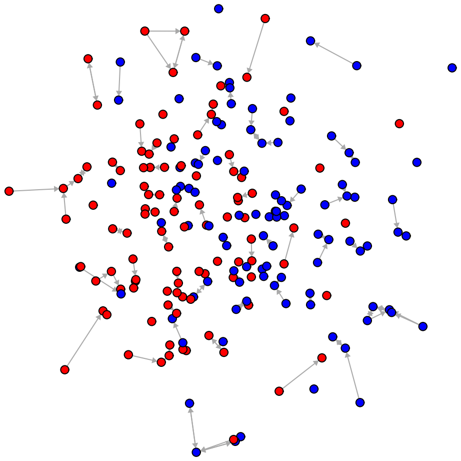

Fig. 4: Network constructed using the non-parametric KAOG technique on a data set of two mixed Gaussians.

IV. Experimental results

The proposed dimensionality reduction technique was evaluated by using the KAOG network formation method as described in Sec. III. The proposed technique was also compared to other two well-known network formation methods, k-NN and ϵ-radius. After the dimensionality reduction step, the projected data set was classified by using the 1-nearest-neighbor classification rule. In the experiments, we applied the techniques to the handwritten digits recognition. The data set used was the *Binary Alphadigits* available online[1]. The data set contains binary 20x16 digits of “0” through “9” and capital “A” through “Z”, with 39 samples (images) of each class. Figure 5 shows some sample images of this data set. In the simulations, each image was mapped as a vertex into an underlying a network.

The parameter optimization was done as follows. For the k-NN network formation technique, parameter k was optimized

[1] http://www.cs.nyu.edu/~roweis/data.html

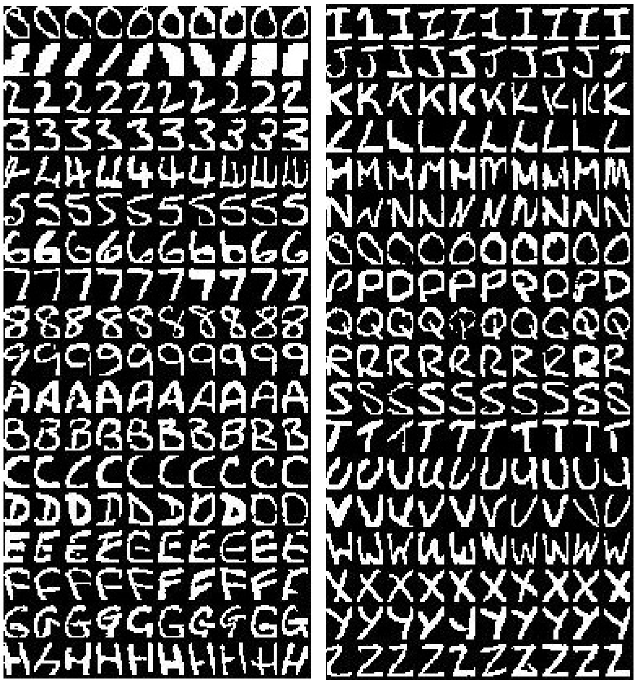

Fig. 5: Image samples from the *Binary Alphadigits* data set[1].

in the interval from 1 to the number of instances of the largest class in the training data set. For the ϵ-radius network formation method, parameter ϵ was optimized in the interval $\{5\%, 10\%, \ldots, 100\%\}$, concerning the average distance among instances in the training data set. The KAOG method is non-parametric. Each experiment was performed by using a 10-fold stratified cross-validation process [16]. In this process, the data set is split in 10 disjoint sets and, in each run, 9 sets are used as training data and 1 set is used as the test data, resulting in a total of 10 runs. The results are averaged over 10 runs, totaling $10 \times 10 = 100$ runs.

Initially, we performed a preliminary experiment using only the images of numbers. The goal was to evaluate the potential of our technique in comparison to another algorithms. Figure 6 shows the results. It can be seen that the KAOG embedding dimensionality reduction technique outperformed both k-NN and ϵ-radius techniques. Also, the KAOG technique is better comparing to the classification using the original image features. For example, when using the first 150 transformed image features, out of 320, the KAOG technique achieved an accuracy around of 80%, against 75%, 74% and 43% when using the original features, the ϵ-radius and the k-NN respectively. These preliminary results show that the proposed technique is promising.

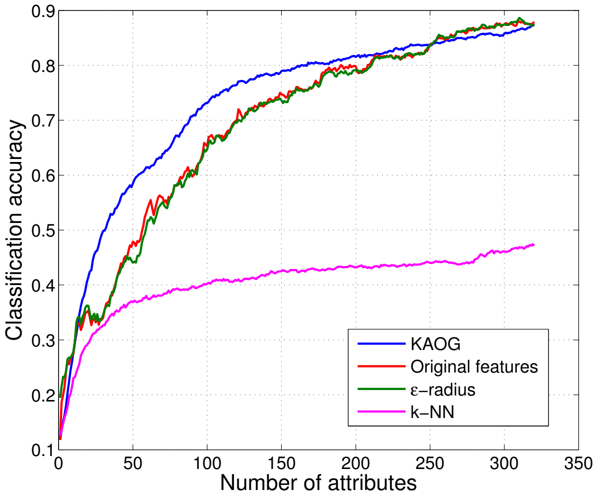

Fig. 6: Classification accuracy on images of numbers from *Binary Alphadigits* data set in function of the number of transformed attributes used in classification.

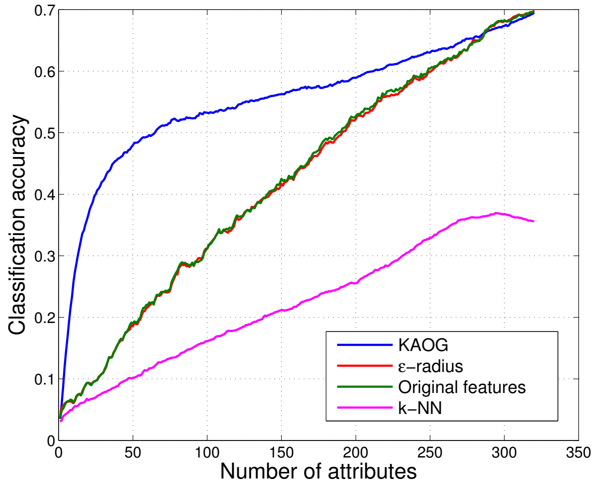

Fig. 7: Classification accuracy on all images available into *Binary Alphadigits* data set in function of the number of transformed attributes used in classification. By using the KAOG network formation, the accuracy increased when using just a small number of transformed features.

In the next experiment, the techniques were analyzed on all images (numbers + letters) from *Binary Alphadigits* data set, resulting in a data set size of 1014 images. Despite the higher complexity of the data set, the KAOG technique was able to perform well, according to Figure 7. Again, one can see that the KAOG embedding dimensionality reduction technique outperformed the other techniques, including the classification using the original image features. For example, when using the first 100 transformed image features, out of 320, the KAOG technique achieved an accuracy around of 54%, against 33%, 33% and 17% when using the original features, the ϵ-radius and the k-NN respectively. These results have showed our technique based on KAOG network can be applied to dimensionality reduction problem with good results in the considered data sets.

V. Conclusion

We have studied the usage of a modified version of the recently proposed k-Associated Optimal Graph (KAOG) to perform supervised dimensionality reduction on image data sets. The proposed technique results in two adjacency matrices which represent the information of input data about both intra-class and inter-class connections. Both matrices are used into a graph embedding framework which is optimized in terms of a projection vector. Experimental studies have showed that the proposed technique achieves competitive results compared to some other classical network formation methods. It has been shown that our technique enhance image processing by reducing the feature dimension and also increased the classification accuracy.

Acknowledgment

The authors would like to acknowledge the São Paulo State Research Foundation (FAPESP) - grant numbers 2009/02036-7 and 2012/07926-3 - and the Brazilian National Council for Scientific and Technological Development (CNPq) for the financial support given to this research.

References

[1] R. O. Duda, D. G. Stork, and P. E. Hart, *Pattern classification*, 2nd ed. Wiley-Interscience, 2000.

[2] H. Cai, K. Mikolajczyk, and J. Matas, “Learning linear discriminant projections for dimensionality reduction of image descriptors,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 2, pp. 338–352, 2011.

[3] S. Chen and D. Zhang, “Semisupervised dimensionality reduction with pairwise constraints for hyperspectral image classification,” *Geoscience and Remote Sensing Letters, IEEE*, vol. 8, no. 2, pp. 369–373, 2011.

[4] B. Raducanu and F. Dornaika, “A supervised non-linear dimensionality reduction approach for manifold learning,” *Pattern Recognition*, vol. 45, no. 6, pp. 2432 – 2444, 2012.

[5] B.-D. Liu, Y.-X. Wang, Y.-J. Zhang, and B. Shen, “Learning dictionary on manifolds for image classification,” *Pattern Recognition*, vol. 46, no. 7, pp. 1879 – 1890, 2013.

[6] I. T. Jolliffe, *Principal Component Analysis*. New York: Springer-Verlag, 1986.

[7] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. Academic Press, 1991.

[8] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, “Graph embedding and extensions: A general framework for dimensionality reduction,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 1, pp. 40–51, 2007.

[9] T. H. Cupertino, J. Huertas, and L. Zhao, “Data clustering using controlled consensus in complex networks,” *Neurocomputing*, no. 0, pp. –, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231213003160

[10] M. G. Carneiro, J. L. Rosa, A. A. Lopes, and L. Zhao, “Network-based data classification: Combining k-associated optimal graphs and high level prediction,” *Journal of the Brazilian Computer Society)*, pp. 1–12, 2013.

[11] T. H. Cupertino, T. C. Silva, and L. Zhao, “Classification of multiple observation sets via network modularity,” *Neural Computing and Applications*, pp. 1–7, 2012. [Online]. Available: http://dx.doi.org/10.1007/s00521-012-1115-y

[12] T. Silva, L. Zhao, and T. H. Cupertino, “Handwritten data clustering using agents competition in networks,” *Journal of Mathematical Imaging and Vision*, pp. 1–13, 2012. [Online]. Available: http://dx.doi.org/10.1007/s10851-012-0353-z

[13] L. Zhao, T. H. Cupertino, and J. R. B. Jr., “Chaotic synchronization in general network topology for scene segmentation,” *Neurocomputing*, vol. 71, no. 16-18, pp. 3360–3366, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231208003111

[14] J. R. Bertini, L. Zhao, R. Motta, and A. de Andrade Lopes, “A nonparametric classification method based on k-associated graphs,” *Information Sciences*, vol. 181, pp. 5435–5456, 2011.

[15] M. Belkin and P. Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation,” *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.

[16] J.-H. Kim, “Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap,” *Computational Statistics and Data Analysis*, vol. 53, pp. 3735–3745, 2009.