A Boosting Framework for Visuality-Preserving Distance Metric Learning and Its Application to Medical Image Retrieval

Liu Yang, *Student Member*, *IEEE*, Rong Jin, Lily Mummert, *Member*, *IEEE*, Rahul Sukthankar, *Member*, *IEEE*, Adam Goode, *Member*, *IEEE*, Bin Zheng, Steven C.H. Hoi, *Member*, *IEEE*, and Mahadev Satyanarayanan, *Fellow*, *IEEE*

Abstract—Similarity measurement is a critical component in content-based image retrieval systems, and learning a good distance metric can significantly improve retrieval performance. However, despite extensive study, there are several major shortcomings with the existing approaches for distance metric learning that can significantly affect their application to medical image retrieval. In particular, "similarity" can mean very different things in image retrieval: resemblance in visual appearance (e.g., two images that look like one another) or similarity in semantic annotation (e.g., two images of tumors that look quite different yet are both malignant). Current approaches for distance metric learning typically address only one goal without consideration of the other. This is problematic for medical image retrieval where the goal is to assist doctors in decision making. In these applications, given a query image, the goal is to retrieve similar images from a reference library whose semantic annotations could provide the medical professional with greater insight into the possible interpretations of the query image. If the system were to retrieve images that did not look like the query, then users would be less likely to trust the system; on the other hand, retrieving images that appear superficially similar to the query but are semantically unrelated is undesirable because that could lead users toward an incorrect diagnosis. Hence, learning a distance metric that preserves both visual resemblance and semantic similarity is important. We emphasize that, although our study is focused on medical image retrieval, the problem addressed in this work is critical to many image retrieval systems. We present a boosting framework for distance metric learning that aims to preserve both visual and semantic similarities. The boosting framework first learns a binary representation using side information, in the form of labeled pairs, and then computes the distance as a weighted Hamming distance using the learned binary representation. A boosting algorithm is presented to efficiently learn the distance function. We evaluate the proposed algorithm on a mammographic image reference library with an Interactive Search-Assisted Decision Support (ISADS) system and on the medical image data set from ImageCLEF. Our results show that the boosting framework compares favorably to state-of-the-art approaches for distance metric learning in retrieval accuracy, with much lower computational cost. Additional evaluation with the COREL collection shows that our algorithm works well for regular image data sets.

Index Terms—Machine learning, image retrieval, distance metric learning, boosting.

1 Introduction

Today, medical diagnosis remains both art and science. Doctors draw upon both experience and intuition, using analysis and heuristics to render diagnoses [1]. When doctors augment personal expertise with research, the medical literature is typically indexed by disease rather than by relevance to current case. The goal of interactive search-assisted decision support (ISADS) is to enable doctors to make better decisions about a given case by retrieving a selection of similar annotated cases from large medical image repositories.

A fundamental challenge in developing such systems is the identification of similar cases, not simply in terms of superficial image characteristics, but in a medically relevant sense. This involves two tasks: extracting a representative set of features and identifying an appropriate measure of similarity in the high-dimensional feature space. The former has been an active research area for several decades. The latter, largely ignored by the medical imaging community, is the focus of this paper.

In an ISADS system, each case maps to a point in a high-dimensional feature space and similar cases to the current case (query) correspond to near neighbors in this space. The neighborhood of a point is defined by a distance metric, such as the euclidean distance. Our previous work showed that the choice of distance metric affects the accuracy of an ISADS system and that machine learning enables the construction of effective domain-specific distance metrics [2]. In a learned distance metric, data points with the same labels (e.g.,

- *L. Yang is with Machine Learning Department, School of Computer Science, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15231. E-mail: liuy@cs.cmu.edu.*
- *R. Jin is with the Department of Computer Science and Engineering, 3115 Engineering Building, Michigan State University, East Lansing, MI 48824. E-mail: rongjin@cse.msu.edu.*
- *L. Mummert and R. Sukthankar are with Intel Research, 4720 Forbes Ave., Suite 410, Pittsburgh, PA 15213. E-mail: lily.b.mummert@intel.com, rahuls@cs.cmu.edu.*
- *A. Goode and M. Satyanarayanan are with the School of Computer Science, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15231. E-mail: agoode@andrew.cmu.edu, satya@cs.cmu.edu.*
- *B. Zheng is with the Department of Radiology, University of Pittsburgh Medical Center, Pittsburgh, PA 15213. E-mail: zengb@upmc.edu.*
- *S.C.H. Hoi is with the Division of Information Systems, School of Computer Engineering, Nanyang Technological University, Singapore 639798. E-mail: chhoi@ntu.edu.sg.*

Manuscript received 1 Jan. 2008; revised 9 Aug. 2008; accepted 24 Oct. 2008; published online 10 Nov. 2009.
Recommended for acceptance by J. Matas.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-01-0001.
Digital Object Identifier no. 10.1109/TPAMI.2008.273.

malignant masses) are closer than data points with different labels (e.g., malignant versus benign). Thus, the labels of the near neighbors of the query are likely to be informative.

1.1 Distance Metric Learning with Side Information

Research in distance metric learning is driven by the need to find meaningful low-dimensional manifolds that capture the intrinsic structure of high-dimensional data. Distance metric learning has been successfully applied to a variety of applications, such as content-based image retrieval [3] and text categorization [4].

Most distance metric learning techniques can be classified into two categories: unsupervised distance metric learning and supervised distance metric learning. The former aims to construct a low-dimensional manifold where geometric relationships between most data points are largely preserved. Supervised distance metric learning makes use of class-label information and identifies the dimensions that are most informative to the classes of examples. A brief overview of the related work is provided in Section 2.

Learning an effective distance metric with side information has recently attracted increasing interest. Typically, the side information is cast in the form of pairwise constraints between data elements, and the goal is to identify features that are maximally consistent with these constraints. In general, there are two types of pairwise constraints: 1) equivalence constraints specifying that the two given elements belong to the same class and 2) inequivalence constraints indicating that the given elements are from different classes. The optimal distance metric is learned by keeping the data elements of equivalence constraints close to each other while separating the data elements of inequivalence constraints apart. A number of approaches have been developed to learn distance metrics from the pairwise constraints. We refer to Section 2 for a comprehensive review.

One of the key challenges in learning a distance metric is its computational cost. This is because many approaches are designed to learn a full matrix of distance metrics whose size scales with the square of the data dimension. In addition to its large size, the requirement that the metric matrix be positive semidefinite further increases the computational cost [5]. Although several algorithms have been proposed to improve the computational efficiency (e.g., [6]), they still tend to be computationally prohibitive when the number of dimensions is large. To address the computational issue, we propose a boosting framework that can efficiently learn distance metrics for high-dimensional data.

1.2 Semantic Relevance and Visual Similarity

Most distance metric learning algorithms aim to construct distance functions that are consistent with the given pairwise constraints. Since these constraints are usually based on the semantic categories of the data, the learned distance metric essentially preserves only the semantic relevance among data points. Thus, a drawback with these approaches is that, when they are applied to image retrieval problems, images ranked at the top of a retrieval list may not be visually similar to the query image, due to the gap between semantic relevance and visual similarity. For instance, a doughnut and a tire have similar shapes, yet belong to different concept categories; a solar car looks almost nothing like a regular car, though functionally, they both belong to the same object category. Since, in image retrieval applications, most users seek images that are both semantically and visually close to the query image, this requires learning distance functions that preserve both semantic relevance and visual resemblance. This issue is of particular importance in medical image retrieval. If the system were to retrieve images that did not look like the query, then doctors would be less likely to trust the system; on the other hand, retrieving images that appear superficially similar to the query but are semantically unrelated is undesirable because that could lead doctors toward an incorrect diagnosis.

We address the challenge by automatically generating links that pair images with high visual resemblance. These visual pairs, together with the provided side information, are used to train a distance function that preserves both visual similarity and semantic relevance between images. The trade-off between semantic relevance and visual similarity can be easily adjusted by the number of visual pairs. A detailed discussion of how these visual pairs are generated is given in Section 4.

The remaining paper is organized as follows: Section 2 reviews the work related to ISADS, distance metric learning, and boosting. Section 3 describes the boosting framework for distance metric learning. Section 4 presents the application of the proposed algorithm to retrieval of both medical images and regular images.

2 Related Work

Over the last decade, the increasing availability of powerful computing platforms and high-capacity storage hardware has driven the creation of large, searchable image databases, such as digitized medical image reference libraries. These libraries have been used to train and validate computer-aided diagnosis (CAD) systems in a variety of medical domains, including breast cancer. However, the value of CAD in clinical practice is controversial, due to their "black-box" nature and lack of reasoning ability [7], [8], [9], [10], [11], despite significant recent progress [12], [13], [14], [15], [16], [17], [18], [19], [20] both in automated detection and characterization of breast masses. An alternative approach, espoused by efforts such as ISADS [2], eschews automated diagnosis in favor of providing medical professionals with additional context about the current case that could enable them to make a more informed decision. This is done by retrieving medically relevant cases from the reference library and displaying their outcomes. Earlier work [2] has demonstrated that learning domain-specific distance metrics significantly improves the quality of such searches.

In general, methods for distance metric learning fall into two categories: supervised and unsupervised learning. Since our work is most closely related to supervised distance metric learning, we omit the discussion of unsupervised distance metric learning and refer readers to a recent survey [21].

In supervised distance metric learning, most algorithms learn a distance metric from a set of equivalence constraints and inequivalence constraints between data objects. The optimal distance metric is found by keeping objects in equivalence constraints close and objects in inequivalence constraints well separated. Xing et al. [22] formulate distance metric learning into a constrained convex programming problem by minimizing the distance between the data points in the same classes under the constraint that the data points from different classes are well separated. This

algorithm is extended to the nonlinear case by the introduction of kernels [23]. Local Linear Discriminative Analysis [24] estimates a local distance metric using the local linear discriminant analysis. Relevant Components Analysis (RCA) [25] learns a global linear transformation from the equivalence constraints. Discriminative Component Analysis (DCA) and Kernel DCA [26] improve RCA by exploring inequivalence constraints and capturing nonlinear transformation via contextual information. Local Fisher Discriminant Analysis (LFDA) [27] extends classical LDA to the case when the side information is in the form of pairwise constraints. Kim et al. [28] provide an efficient incremental learning method for LDA, by adopting sufficient spanning set approximation for each update step. Schultz and Joachims [29] extend the support vector machine to distance metric learning by encoding the pairwise constraints into a set of linear inequalities. Neighborhood Component Analysis (NCA) [30] learns a distance metric by extending the nearest neighbor classifier. The maximum-margin nearest neighbor (LMNN) classifier [6] extends NCA through a maximum margin framework. Yang et al. [31] propose a Local Distance Metric (LDM) that addresses multimodal data distributions in distance metric learning by optimizing local compactness and local separability in a probabilistic framework. Finally, a number of recent studies [28], [32], [33], [34], [35], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49] focus on examining and exploring the relationship among metric learning, dimensionality reduction, kernel learning, semi-supervised learning, and Bayesian learning.

Learning distance metrics by a boosting framework was first presented by Hertz et al. [50], [51]. In addition, in [36], [37], [52], different boosting strategies are presented to learn distance functions from labeled data. Although all of these algorithms employ a boosting strategy to learn a distance function, our algorithm differs from the existing work in that earlier algorithms for distance function learning closely follow AdaBoost [53] without considering the optimization of the specified objective functions. Some of the existing methods (e.g., [52]) do not have a well-specified objective function; therefore, the convergence of their algorithms and the optimality of the resulting distance function are unclear. In contrast, our algorithm is based on the optimization of the objective function specified in our study. Our contributions include a theoretical analysis about the convergence condition of our algorithm and the optimality of the resulting distance function. We believe that the theoretical analysis of the proposed algorithm is important and could be instrumental to the performance of our boosting framework.

We would also like to mention some recent developments in nonmetric distance learning, such as Generalized Nonmetric Multidimensional Scaling [54]. Although nonmetric distance learning appears to be more flexible than metric distance learning, we believe that metric distance, in general, is not only more intuitive but also more robust to data noise due to the constraints imposed by the triangle inequality.

3 A Boosting Framework for Distance Metric Learning

In this section, we present a novel boosting framework, termed **BDM** (we follow the terminology from [2]), that automatically learns a distance function from a given set of pairwise constraints. The main idea is to iteratively generate a set of binary features from the side information. The learned binary features are used for data representation, and the distance is computed as a weighted Hamming distance based on the learned binary data representation.

3.1 Preliminaries

We denote by $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ the collection of data points. Each $\mathbf{x} \in \mathbf{R}^d$ is a vector of d dimensions. We denote by $X = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ the data matrix containing the input features of both the labeled and the unlabeled examples. Following [22], we assume that the side information is available in the form of labeled pairs, i.e., whether or not two examples are in the same semantic category or not. For convenience of discussion, below we refer to examples in the same category as "similar" examples and examples in different categories as "dissimilar" examples. Let the set of labeled example pairs be denoted by

$$\mathcal{P} = \{(\mathbf{x}_i, \mathbf{x}_j, y_{i,j}) | \mathbf{x}_i \in \mathcal{D}, \mathbf{x}_j \in \mathcal{D}, y_{i,j} \in \{-1, 0, +1\}\},$$

where the class label $y_{i,j}$ is positive (i.e., $+1$) when $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar, and negative (i.e., -1) when $\mathbf{x}_i$ and $\mathbf{x}_j$ are different. $y_{i,j}$ is set to zero when the example pair $(\mathbf{x}_i, \mathbf{x}_j)$ is unlabeled. Finally, we denote by $d(\mathbf{x}_i, \mathbf{x}_j)$ the distance function between $\mathbf{x}_i$ and $\mathbf{x}_j$. Our goal is to learn a distance function that is consistent with the labeled pairs in $\mathcal{P}$.

Remark 1. Note that standard labeled examples can always be converted into a set of labeled pairs by assigning two data points from the same category to the positive class and two data points from different categories to the negative class. Similar pairwise class labels are commonly employed in multiclass multimedia retrieval applications [55], [56]. It is important to emphasize that the reverse is typically difficult, i.e., it is usually difficult to infer the unique category labels of examples from the labeled pairs [57].

Remark 2. We label two images in the training set as similar if they either match in semantic category or if they appear visually related, as our goal is to simultaneously preserve both the semantic relevance as well as the visual similarity. For instance, two images could be defined to be similar only if they belonged to the same semantic category or similarity could be defined based on the images' visual similarity according to human perception.

3.2 Definition of Distance Function

Before presenting the boosting algorithm, we need to define a distance function $d(\mathbf{x}_i, \mathbf{x}_j)$ that is nonnegative and satisfies the triangle inequality. A typical definition of distance function used by several distance metric learning algorithms (e.g., [22], [31]) is

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^\top A (\mathbf{x}_i - \mathbf{x}_j)}, \quad (1)$$

where $A \in \mathbb{R}^{d\times d}$ is a positive semidefinite matrix that specifies the distance metric. One drawback with the definition in (1) arises from its high computational cost due to the size of A and the constraint that matrix A has to be positive semidefinite. This is observed in our empirical study. When the dimensionality $d = 500$, we find that estimating A in (1) is computationally very expensive.

In order to address the above problems, we present here a nonlinear distance function defined as follows:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{T} \alpha_t (f_t(\mathbf{x}_i) - f_t(\mathbf{x}_j))^2, \tag{2}$$

where each $f(\mathbf{x}) : \mathbb{R}^n \to \{-1, +1\}$ is a binary classification function (note that we define the image of the binary f to be $\{-1, +1\}$ instead of $\{0, 1\}$ for a more concise presentation below) and $\alpha_t > 0$, $t = 1, 2, \ldots, T$, are the combination weights. The key idea behind the above definition is to first generate a binary representation $(f_1(\mathbf{x}), \ldots, f_T(\mathbf{x}))$ by applying the classification function $\{f_i(\mathbf{x})\}_{i=1}^{T}$ to $\mathbf{x}$. Then, the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ is computed as a weighted Hamming distance between the binary representations of the two examples. Compared to (1), (2) is advantageous in that it allows for a nonlinear distance function. Furthermore, the iterative updates of the binary data representation, and consequently, the distance function, are the key to the efficient algorithm that is presented in the next section. We emphasize that although (2) appears to be similar to the distance function defined in [36], [37], it differs from the existing work in that each binary function takes into account all of the features. In contrast, each binary function in [36], [37] is limited to a single feature and therefore is significantly less general than the proposed algorithm.

The following theorem shows that the distance function defined in (2) is indeed a pseudometric, i.e., satisfies all the conditions of a distance metric except for $d(\mathbf{x}, \mathbf{y}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{y}$. More specifically, we have the following theorem:

Theorem 3.1. *The distance function defined in (2) satisfies all the properties of a pseudometric, i.e., 1) $d(\mathbf{x}_i, \mathbf{x}_j) = d(\mathbf{x}_j, \mathbf{x}_i)$, 2) $d(\mathbf{x}_i, \mathbf{x}_j) \geq 0$, and 3) $d(\mathbf{x}_i, \mathbf{x}_j) \leq d(\mathbf{x}_i, \mathbf{x}_k) + d(\mathbf{x}_k, \mathbf{x}_j)$.*

The first and second properties are easy to verify. To prove the third property, i.e., the triangle inequality, in Theorem 3.1, we need the following lemma:

Lemma 3.2. *The following inequality:*

$$(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 \leq (f(\mathbf{x}_i) - f(\mathbf{x}_k))^2 + (f(\mathbf{x}_k) - f(\mathbf{x}_j))^2 \tag{3}$$

holds for any binary function $f : \mathbb{R}^d \to \{-1, +1\}$.

The proof of the above lemma can be found in Appendix A. It is straightforward to show the triangle inequality in Theorem 3.1 using Lemma 3.2 since $d(\mathbf{x}_i, \mathbf{x}_j)$ is a linear combination of $(f_k(\mathbf{x}_i) - f_k(\mathbf{x}_j))^2$.

3.3 Objective Function

The first step toward learning a distance function is to define an appropriate objective function. The criterion employed by most distance metric learning algorithms is to identify a distance function $d(\mathbf{x}_i, \mathbf{x}_j)$ that gives a small value when $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar and a large value when they are different. We can generalize this criterion by stating that, for any data point, its distance to a similar example should be significantly smaller than the distance to an example that is not similar. This generalized criterion is cast into the following objective function, i.e.,

$$\begin{aligned} err(\mathcal{P}) = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} & I[y_{i,j} = -1] I[y_{i,k} = +1] \\ & I[d(\mathbf{x}_i, \mathbf{x}_j) > d(\mathbf{x}_i, \mathbf{x}_k)], \end{aligned} \tag{4}$$

where the indicator $I[x]$ outputs 1 when the Boolean variable x is true and zero otherwise. In the above, we use $I[y_{i,j} = -1]$ to select the pairs of dissimilar examples, and $I[y_{i,k} = +1]$ to select the pairs of similar examples. Every triple $(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k)$ is counted as an error when the distance between the similar pair $(\mathbf{x}_i, \mathbf{x}_j)$ is larger than the distance between the dissimilar pair $(\mathbf{x}_i, \mathbf{x}_k)$. Hence, the objective function $err(\mathcal{P})$ essentially measures the number of errors when comparing the distance between a pair of similar examples to the distance between a pair of dissimilar examples.

Although the classification error $err(\mathcal{P})$ seems to be a natural choice for the objective function, it has two shortcomings when used to learn a distance function.

- It is well known in the study of machine learning that directly minimizing the training error tends to produce a model that overfits the training data.
- The objective function $err(\mathcal{P})$ is a nonsmooth function due to the indicator function $I[d(\mathbf{x}_i, \mathbf{x}_j) > d(\mathbf{x}_i, \mathbf{x}_k)]$ and therefore is difficult to optimize.

To overcome the shortcomings of $err(\mathcal{P})$, we propose the following objective function for distance metric learning:

$$\begin{aligned} F(\mathcal{P}) = \sum_{i,j,k=1}^{n} & I[y_{i,j} = -1] I[y_{i,k} = +1] \\ & \times \exp\big(d(\mathbf{x}_i, \mathbf{x}_k) - d(\mathbf{x}_i, \mathbf{x}_j)\big). \end{aligned} \tag{5}$$

The key difference between $F(\mathcal{P})$ and $err(\mathcal{P})$ is that $I[d(\mathbf{x}_i) > d(\mathbf{x}_j)]$ is replaced with $\exp\big(d(\mathbf{x}_i, \mathbf{x}_k) - d(\mathbf{x}_i, \mathbf{x}_j)\big)$. Since $\exp\big(d(\mathbf{x}_i, \mathbf{x}_k) - d(\mathbf{x}_i, \mathbf{x}_j)\big) > I[d(\mathbf{x}_i) > d(\mathbf{x}_j)]$, by minimizing the objective function $F(\mathcal{P})$, we are able to effectively reduce the classification error $err(\mathcal{P})$. The advantages of using $\exp\big(d(\mathbf{x}_i, \mathbf{x}_k) - d(\mathbf{x}_i, \mathbf{x}_j)\big)$ versus $I[d(\mathbf{x}_i) > d(\mathbf{x}_j)]$ are twofold.

- Since $\exp(d(\mathbf{x}_i, \mathbf{x}_k) - d(\mathbf{x}_i, \mathbf{x}_j))$ is a smooth function, the objective function $F(\mathcal{P})$ can, in general, be minimized effectively using standard optimization techniques.
- Similarly to AdaBoost [58], by minimizing the exponential loss function in $F(\mathcal{P})$, we are able to maximize the classification margin and therefore reduce the generalized classification error according to [53].

Despite the advantages stated above, we note that the number of terms in (5) is on the order of $\mathcal{O}(n^3)$, potentially creating an expensive optimization problem. This observation motivates the development of a computationally efficient algorithm.

3.4 Optimization Algorithm

Given the distance function in (2), our goal is to learn appropriate classifiers $\{f_t(\mathbf{x})\}_{t=1}^T$ and combination weights $\{\alpha_t\}_{t=1}^T$. In order to efficiently learn the parameters and functions, we follow the idea of boosting and adopt a greedy approach for optimization. More specifically, we start with a constant function for distance, i.e., $d_0(\mathbf{x}_i, \mathbf{x}_j) = 0$, and learn a distance function $d_1(\mathbf{x}_i, \mathbf{x}_j) = d_0(\mathbf{x}_i, \mathbf{x}_j) + \alpha_1(f_1(\mathbf{x}_i) - f_1(\mathbf{x}_j))^2$. Using this distance function, the objective function in (5) becomes a function of α_1 and $f_1(\mathbf{x})$, and can be optimized efficiently using bound optimization [59] as described later. In general, given a distance function $d_{t-1}(\mathbf{x}_i, \mathbf{x}_j)$ that is learned in iteration $t-1$, we learn α_t and $f_t(\mathbf{x})$ by using the following distance function $d_t(\mathbf{x}_i, \mathbf{x}_j)$:

$$d_t(\mathbf{x}_i, \mathbf{x}_j) = d_{t-1}(\mathbf{x}_i, \mathbf{x}_j) + \alpha_t\big(f_t(\mathbf{x}_i) - f_t(\mathbf{x}_j)\big)^2.$$

Using the above expression for distance function, the objective function at iteration t, denoted by $F_t(\mathcal{P})$, in (5) becomes a function of α_t and $f_t(\mathbf{x})$, i.e.,

$$\begin{aligned} F_t(\mathcal{P}) = \sum_{i,j,k=1}^{n} & I[y_{i,j} = -1] I[y_{i,k} = +1] \\ & \times \exp(d_{t-1}(\mathbf{x}_i, \mathbf{x}_k) - d_{t-1}(\mathbf{x}_i, \mathbf{x}_j)) \\ & \times \exp(\alpha_t[(f_t(\mathbf{x}_i) - f_t(\mathbf{x}_k))^2 - (f_t(\mathbf{x}_i) - f_t(\mathbf{x}_j))^2]). \end{aligned}$$

To simplify our expression, we introduce the following notations:

$$d_{i,j} \equiv d_{t-1}(\mathbf{x}_i, \mathbf{x}_j), \tag{6}$$

$$f_i \equiv f_t(\mathbf{x}_i), \tag{7}$$

$$\phi_{i,j}^{\pm} \equiv I[y_{i,j} = \pm 1] \exp(\pm d_{i,j}). \tag{8}$$

Using the above notations, $F(\mathcal{P})$ is expressed as follows:

$$F_t(\mathcal{P}) = \sum_{i,j,k=1}^{n} \phi_{i,j}^{-} \phi_{i,k}^{+} \exp(\alpha_t(f_i - f_k)^2 - \alpha_t(f_i - f_j)^2). \tag{9}$$

Hence, the key question is how to find the classifier $f(\mathbf{x})$ and α that minimizes the objective function in (9). For convenience of discussion, we drop the index t for α_t and $f_t(\mathbf{x})$, i.e., $\alpha_t \to \alpha$ and $f_t(\mathbf{x}) \to f(\mathbf{x})$. Now, we apply the bound optimization algorithm [59] to optimize $F_t(\mathcal{P})$ with respect to α and $f(\mathbf{x})$. The main idea is to approximate the difference between the objective functions of the current iteration and the previous iteration by a convex upper bound that has a closed-form solution. As shown in [59], the bound optimization is guaranteed to find a local optimal solution.

Like most bound optimization algorithms, instead of minimizing $F(\mathcal{P})$ in (9), we will minimize the difference between objective functions from two consecutive iterations, i.e.,

$$\Delta(\alpha, \mathbf{f}) = F_t(\mathcal{P}) - F_{t-1}(\mathcal{P}), \tag{10}$$

where $\mathbf{f} = (f_1, \ldots, f_n)$ and $F_{t-1}(\mathcal{P}) = \sum_{i,j,k=1}^{n} \phi_{i,j}^{-} \phi_{i,k}^{+}$ is the objective function of the first $t-1$ iterations. Note that $\Delta(\alpha, \mathbf{f}) = 0$ when $\alpha = 0$. This condition guarantees that when we minimize $\Delta(\alpha, \mathbf{f})$, the resulting $F_t(\mathcal{P})$ is smaller than $F_{t-1}(\mathcal{P})$, and therefore, the objective function will monotonically decrease through iterations. In addition, as shown in [59], minimizing the bound is guaranteed to find a locally optimal solution.

First, in the following lemma, we construct an upper bound for $\Delta(\alpha, \mathbf{f})$ that decouples the interaction between α and $\mathbf{f}$. Before stating the result, we introduce the concept of a "graph Laplacian" for readers who may not be familiar with the term. A graph Laplacian for a similarity matrix S, denoted by $L(S)$, is defined as $L = \mathrm{diag}(S\mathbf{1}) - S$, where $\mathbf{1}$ is an all-one vector and operator $\mathrm{diag}(\mathbf{v})$ turns vector $\mathbf{v}$ into a diagonal matrix.

Lemma 3.3. *For any $\alpha > 0$ and binary vector $\mathbf{f} \in \{-1, +1\}^n$, the following inequality holds:*

$$\Delta(\alpha, \mathbf{f}) \le \frac{\exp(-8\alpha) - 1}{8} \mathbf{f}^\top L^+ \mathbf{f} + \frac{\exp(8\alpha) - 1}{8} \mathbf{f}^\top L^- \mathbf{f}, \tag{11}$$

where L^- and L^+ are the graph Laplacians for the similarity matrices S^- and S^+, respectively, defined as

$$S_{i,j}^- = \frac{1}{2} \phi_{i,j}^+ (\mu_i^- + \mu_j^-), \; S_{i,j}^+ = \frac{1}{2} \phi_{i,j}^- (\mu_i^+ + \mu_j^+), \tag{12}$$

where $\mu_i^{\pm}$ is defined as

$$\mu_i^{\pm} = \sum_{j=1}^{n} \phi_{i,j}^{\pm}. \tag{13}$$

Recall that $\phi_{i,j}^{\pm}$ is defined as $\phi_{i,j}^{\pm} = I[y_{i,j} = \pm 1] \exp(\pm d_{i,j})$ in (8). The detailed proof of this lemma is given in Appendix B.

Remark. Since $\phi_{i,j}^+ \propto I[y_{i,j} = 1]$ (8), the similarity S^- depends only on the data points from the must-link pairs (equivalence constraints). Hence, $\mathbf{f}^\top L^- \mathbf{f}$ in (11) essentially measures the inconsistency between the binary vector $\mathbf{f}$ and the must-link constraints. Similarly, $\mathbf{f}^\top L^+ \mathbf{f}$ in (11) measures the inconsistency between $\mathbf{f}$ and the cannot-link pairs (inequivalence constraints). Hence, the upper bound in (11) essentially computes the overall inconsistency between the labeled pairs and the binary vector $\mathbf{f}$.

Next, using Lemma 3.3, we derive additional bounds for $\Delta(\alpha, \mathbf{f})$ by removing α. This result is summarized in the following theorem.

Theorem 3.4. *For any binary vector $\mathbf{f} \in \{-1, +1\}^n$, the following inequality holds:*

$$\min_{\alpha \ge 0} \Delta(\alpha, \mathbf{f}) \le -\frac{1}{8} \left(\max(0, \sqrt{\mathbf{f}^\top L_+ \mathbf{f}} - \sqrt{\mathbf{f}^\top L_- \mathbf{f}}) \right)^2 \tag{14}$$

$$\le -\frac{\big(\max(0, \mathbf{f}^\top L^+ \mathbf{f} - \mathbf{f}^\top L^- \mathbf{f})\big)^2}{8n(\sqrt{\lambda_{\max}(L^-)} + \sqrt{\lambda_{\max}(L^+)})^2}, \tag{15}$$

where $\lambda_{\max}(S)$ is the maximum eigenvalue of matrix S.

The proof of this theorem can be found in Appendix C. In the following discussion, we will focus on minimizing the upper bound of the objective function stated in Theorem 3.4, which allows us to reduce the computational cost dramatically.

In order to search for the optimal binary solution $\mathbf{f}$ that minimizes the upper bound of $\Delta(\alpha, \mathbf{f})$, we decide to first search for a continuous solution for $\mathbf{f}$ and then convert the continuous $\mathbf{f}$ into a binary one by comparing to a threshold b. In particular, we divide the optimization procedure into two steps:

- searching for the continuous $\mathbf{f}$ that minimizes the upper bound in (15) and
- searching for the threshold b that minimizes the upper bound in (14) for a continuous solution $\mathbf{f}$.

To differentiate the continuous solution $\mathbf{f}$, we furthermore denote by $\hat{\mathbf{f}}$ the binary solution. It is important to note that the two steps use different upper bounds in Lemma 3.3. This is because the looser upper bound in (15) allows for efficient computation of continuous solution $\mathbf{f}$, while the tighter upper bound in (11) allows for a more accurate estimation of threshold b.

Finally, the optimization problems related to the two steps are summarized as follows, respectively:

$$\max_{\mathbf{f}\in\mathbb{R}^n} \mathbf{f}^\top (L^+ - L^-)\mathbf{f}, \tag{16}$$

and

$$\begin{aligned} &\max_{b\in\mathbb{R}} \sqrt{\hat{\mathbf{f}}^\top L^+ \hat{\mathbf{f}}} - \sqrt{\hat{\mathbf{f}}^\top L^- \hat{\mathbf{f}}} \\ &\text{s.t. } \hat{f}_i = \begin{cases} 1, & f_i > b, \\ -1, & f_i \le b. \end{cases} \end{aligned} \tag{17}$$

It is clear that the optimal solution to (16) is the maximum eigenvector of matrix $L^+ - L^-$, and therefore, can be computed very efficiently. To find the b that optimizes the problem in (17), it is sufficient to consider $f_1, f_2, \ldots, f_n$, in turn, as the candidate solutions.

Given the optimal $\mathbf{f} = (f_1, \ldots, f_n)$, the next question is how to learn a classification function $f(\mathbf{x})$ to approximate $\mathbf{f}$. Here, we consider two cases: the linear classifier and the nonlinear classifier. In the first case, we assume that the classification function $f(\mathbf{x})$ is based on a linear transformation of $\mathbf{x}$, i.e., $f(\mathbf{x}) = \mathbf{u}^\top \mathbf{x}$, where $\mathbf{u}$ is a projection vector that needs to be determined. Then, the optimization problem in (16) is converted into the following problem:

$$\max_{\mathbf{u}^\top\mathbf{u}=1} \mathbf{u}^\top X(L^+ - L^-)X^\top \mathbf{u}. \tag{18}$$

It is not difficult to see that the optimal projection $\mathbf{u}$ that maximizes (18) is the maximum eigenvector of $X(L^+ - L^-)X^\top$. In the second case, we exploit the "kernel trick." Specifically, we introduce a nonlinear kernel function $k(\mathbf{x}, \mathbf{x}')$ and assume the classification function $f(\mathbf{x})$ as

$$f(\mathbf{x}) = \sum_{i=1}^{n} k(\mathbf{x}_i, \mathbf{x}) u_i.$$

Similarly to the linear case, we calculate the optimal projection $\mathbf{u} = (u_1, \ldots, u_n)$ by computing the maximum eigenvector of $K(L^+ - L^-)K^\top$, where K is a nonlinear kernel similarity matrix with $K_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. Fig. 1 summarizes the proposed boosted distance metric learning algorithm of both the linear and the nonlinear cases.

- Initialize $d_{i,j} = 0$ for any $i, j = 1, 2, \ldots, n$.
- For $t = 1, 2, \ldots, T$
  - Compute S^+ and S^- using Eqn. 12.
  - Compute the graph Laplacian $L(S^+)$ and $L(S^-)$.
  - Compute the maximum eigenvector $\mathbf{u}$ for the eigenvector problem
    $X(L^+ - L^-)X^\top \mathbf{u} = \lambda \mathbf{u}$ (linear case) or
    $K(L^+ - L^-)K\mathbf{u} = \lambda \mathbf{u}$ (nonlinear case).
  - Compute the predicted values $\mathbf{f} = X^\top \mathbf{u}$ (linear case) and $\mathbf{f} = K^\top \mathbf{u}$ (nonlinear case).
  - Find the optimal threshold b, and compute the binary output $\hat{\mathbf{f}}$ as $\hat{\mathbf{f}} = \text{sign}(\mathbf{f} - b)$.
  - Compute the optimal α that maximizes Eqn. 11 by the equation
    $$\alpha = (\log(\mathbf{f}^\top L^+ \mathbf{f}) - \log(\mathbf{f}^\top L^- \mathbf{f}))/16$$
  - Update the distance $d_{i,j} \leftarrow d_{i,j} + \alpha(\hat{f}_i - \hat{f}_j)^2$.

Fig. 1. Distance metric learning algorithm in a boosting framework.

To further ensure that our algorithm is effective in reducing the objective function despite being designed to minimize the upper bound of the objective function, we present the following theorem:

Theorem 3.5. *Let (S_t^+, S_t^-), $t = 1, \ldots, T$ be the similarity matrices that are computed by running the boosting algorithm (in Fig. 1) using (12). Let L_t^+ and L_t^- be the corresponding graph Laplacians. Then, the objective function at the $T+1$ iteration, i.e., $F_{T+1}(\mathcal{P})$, is bounded as follows:*

$$F_{T+1}(\mathcal{P}) \le F_0(\mathcal{P}) \prod_{t=0}^{T} (1 - \gamma_t), \tag{19}$$

where

$$F_0 = \sum_{i,j,k=1}^{n} I[y_{i,j} = -1] I[y_{i,k} = +1],$$

$$\gamma_t = \frac{[\lambda_{\max}(L_t^+ - L_t^-)]^2}{8\lambda_{\max}(S_t^+ + S_t^-)(\lambda_{\max}(L_t^+) + \lambda_{\max}(L_t^-))}.$$

The proof of this theorem can be found in Appendix D. Evidently, we note that γ is bounded between $(0, 1/8)$. As revealed in the above theorem, although we only aim to minimize the upper bound of the objective function, the upper bound of the objective function decreases by a factor of $1 - \gamma_t$ in each iteration, and therefore, the objective function will, in general, decrease rapidly. This claim is supported by our experimental results below.

3.5 Preserving Visual Similarity

As pointed out in Section 1, most distance metric learning algorithms learn a distance metric that only preserves the semantic similarity without taking into account the visual resemblance between images. Fig. 2 shows a pair of two images whose distance is very "small" according to a distance metric learned from the labeled examples. Note that, although both images are malignant according to the medical annotation, their appearances are rather different. By retrieving images that are only medically relevant, the

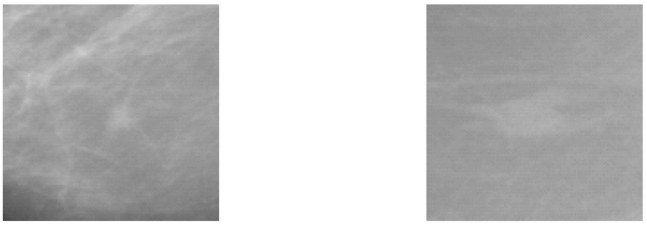


Fig. 2. Two images with the same semantic label (malignant masses in this example) can look very different visually. In an ISADS application, it is important for the system to retrieve examples that are both visually and semantically similar.

system is poorly suited for assisting doctors in providing the necessary context for informed decision making.

To address this problem, we introduce additional pairwise constraints to reflect the requirement of visual similarity. These additional pairwise constraints, referred to as "**visual pairs**," are combined with the equivalence and inequivalence constraints to train a distance metric using the boosting algorithm that is described above. Ideally, the visual pairs would be specified manually by domain experts. However, in the absence of such labels, we represent an image by a vector of visual features and approximate the visual pairs by the pairs of images that fall within a small euclidean distance in the space of visual features. By incorporating the visual pairs as a part of the pairwise constraints, the resulting distance function will reflect not only the semantic relevance among images, but also the visual similarity between images. Furthermore, the trade-off between visual and semantic similarity in learning a distance function can be adjusted by varying the number of visual pairs. As shown in our experiments, employing a large number of visual pairs biases the learned metric toward preserving visual similarity. Finally, we note that the same set of low-level image features is used to assess the medical relevance of images and to generate visual pairs. The key difference is that, in generating visual pairs, every feature is treated with equal importance; in contrast, the semantic relevance between two images is judged by a weighted distance, and therefore, only a subset or combinations of image features determines the semantic relevance of images.

We can also interpret visual pairs from the viewpoint of Bayesian learning. In particular, introducing visual pairs into our learning scheme is essentially equivalent to introducing a Bayesian prior for the target distance function. Note that 1) the same set of visual features is used to judge the semantic relevance and visual similarity and 2) visual pairs are generated by the euclidean distance. Hence, the introduction of visual pairs serves as a regularizer for the distance function to be close to the euclidean distance. We emphasize the importance of regularization in distance metric learning, particularly when the number of pairwise constraints is limited. Since most distance functions involve a large number of parameters, overfitting is likely in the absence of appropriate regularization; resulting distance functions are likely to fit the training data very well, yet will fail to correctly predict the distances between the examples in the testing set. This issue is examined further in our experiments below.

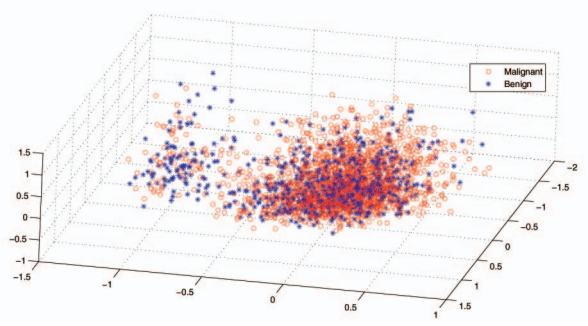


Fig. 3. Three-dimensional PCA representation of the malignant (red) class and benign (blue) class.

4 Applications

This section presents evaluations of the proposed boosting framework for learning distance functions in the context of both medical and nonmedical image retrieval applications. We denote the basic algorithm by **BDM** and the algorithm augmented with automatically generated visual pairs as **BDM+V**. The first set of experiments employs our method in an ISADS application for breast cancer. The ISADS application allows a radiologist examining a suspicious mass in a mammogram to retrieve and study similar masses with outcomes before determining whether to recommend a biopsy. We first describe the image repository used by the application. We then empirically examine and evaluate different properties of the proposed algorithm, including the convergence of the proposed algorithm, the effect of visual pairs on the performance of image retrieval and classification, and the impact of training set size. Finally, we also evaluate the proposed algorithm using the medical image data set from ImageCLEF [60]. To demonstrate **BDM+V**'s generalized efficacy on regular image data sets beyond the medical domain, we also present retrieval and classification results on the standard Corel data set.

4.1 Reference Library: UPMC Mammograms Data Set

We used an image reference library based on digitized mammograms created by the Imaging Research Center of the Department of Radiology at the University of Pittsburgh. The library consists of 2,522 mass regions of interest (ROI) including 1,800 pathology-verified malignant masses and 722 CAD-cued benign masses. Each mass ROI is represented by a vector of 38 morphological and intensity distribution-related features, within which nine features are computed from the whole breast area depicted in the digitized mammogram (global features) and the remaining features are computed from the segmented mass region and its surrounding background tissue (local features). The extracted visual features are further normalized by the mean and the standard deviation computed from the 2,522 selected mass regions in the image data set. A detailed description of the features, the normalization step, and region segmentation are described in [61], [62]. Fig. 3 shows a significant overlap between the two classes in the space spanned by the first three principal eigenvectors computed by Principal Component Analysis (PCA). This result illustrates the difficulty in separating classes using simple methods.

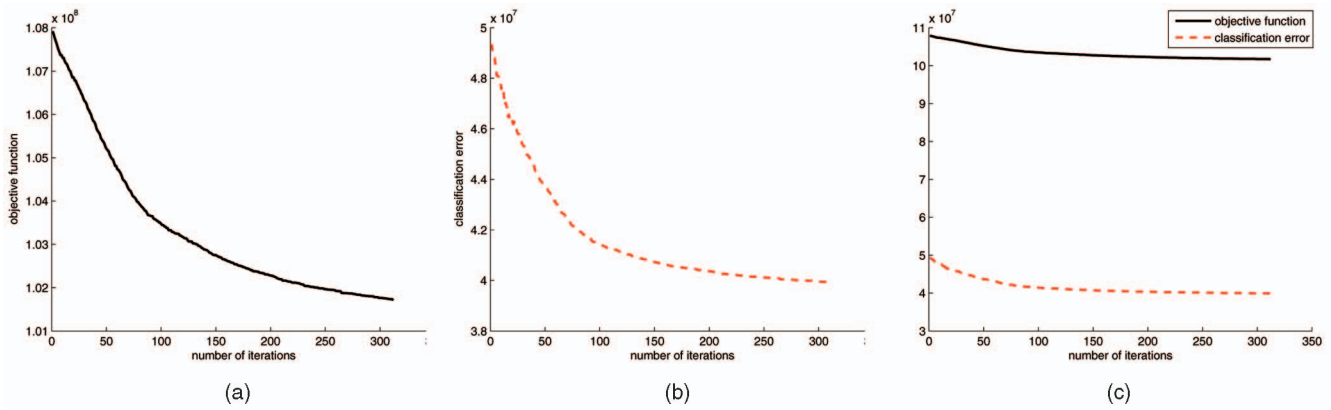


Fig. 4. Reduction of objective function and error rate over iterations (312 iterations in total). (a) Objective function. (b) Error rate $err(\mathcal{P})$. (c) Objective function versus error rate $err(\mathcal{P})$.

4.2 Experiment Setup

We randomly select 600 images from the reference library to serve as the training data set. Among them, 300 images depict malignant masses and 300 depict CAD-generated benign mass regions. The remaining 1,922 images are used for testing. Through these experiments, unless specified, the linear **BDM** (described in Fig. 1) is used for evaluation.

We evaluate the proposed algorithm in the context of ISADS using two metrics. The first metric, classification accuracy, indicates the extent to which malignant images can be detected based on the images that are retrieved by the system [18], [19]. We compute classification accuracy by the K Nearest Neighbor (KNN) classifier: Given a test example $\mathbf{x}$, we first identify the K training examples that have the shortest distance to $\mathbf{x}$, where distance is computed using the metric learned from training examples; we then compute the probability that $\mathbf{x}$ is malignant based on the percentage of its K nearest neighbors that belong to the malignant class. These probabilities for test examples are used to generate the Receiver Operating Characteristic (ROC) curve by varying the threshold of the probability for predicting malignancy. Finally, the classification accuracy is assessed by the area under the ROC curve. As has been pointed out by many studies, the ROC curve is a better metric for evaluating classification accuracy than error rate, particularly when the populations of classes are skewed. Cross validation has indicated that the optimal number of nearest neighbors (i.e., K) in KNN is 10. Every experiment is repeated 10 times with randomly selected training images and the final result is computed as an average over these 10 runs. Both the mean and standard deviation of the area under the ROC curve are reported in the study.

The second metric, retrieval accuracy, reflects the proportion of retrieved images that are medically relevant (i.e., in the same semantic class) to the given query [16], [17]. Unlike classification accuracy where only a single value is calculated, retrieval accuracy is computed as a function of the number of retrieved images and thus provides a more comprehensive picture for the performance of ISADS. We evaluate retrieval accuracy in a leave-one-out manner, i.e., using one medical image in the test data set as the query and the remaining images in the test data set as the gallery when we conduct the experiment of image retrieval. For a given test image, we rank the images in the gallery in the ascending order of their distance to the query image. We define the retrieval accuracy for the ith test query image at rank position k, denoted by $r(\mathbf{q}_i^k)$, as the percentage of the first k ranked images that share the same semantic class (i.e., benign or malignant) as the query image:

$$r(\mathbf{q}_i^k) = \frac{\sum_{j=1}^{k} I[y_i = y_j]}{k}, \tag{20}$$

where j in the summation refers to the indices of the top k ranked images. The overall retrieval accuracy at each rank position is an average over all images in the testing set.

4.3 Empirical Evaluation of the Proposed Algorithm (BDM+V)

In this section, we study the convergence of the proposed algorithm, the performance of the proposed algorithm for both image classification and retrieval, and, furthermore, the effect of visual pairs on image retrieval.

4.3.1 Convergence of the Objective Function

Fig. 4a shows the reduction of the objective function (5) and Fig. 4b shows the reduction of the error rate $err(\mathcal{P})$ in (4), both as a function of the number of iterations. The "number of iterations" in Fig. 4 corresponds to the "T" from (2) and Fig. 1. Recall that the error rate $err(\mathcal{P})$ measures the number of errors when comparing the distance between a pair of similar examples to the distance between a pair of dissimilar examples. We also compare the change of the two in the same figure (see Fig. 4c). The iteration stops when the relative change in the objective function is smaller than a specified threshold (10^{-5} in our study).

First, we clearly observe that the value of the objective function drops at a rapid rate, which confirms the theoretic analysis stated in Theorem 3.5. Second, we observe that the overall error rate is also reduced significantly, and indeed, is upper bounded by the objective function in (5), as discussed in Section 3, although the bound is rather loose.

4.3.2 Effect of Visual Pairs

We first evaluate how the visual pairs affect the retrieval accuracy of BDM. Fig. 5 summarizes the retrieval accuracy of BDM+V and BDM (i.e., with and without using the visual pairs). For the purpose of comparison, we also include the

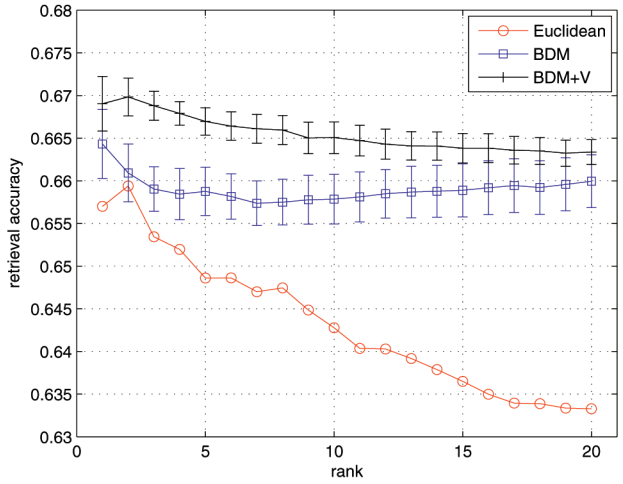


Fig. 5. Comparison of retrieval accuracy. The learned metrics significantly outperform euclidean; adding visual pairs (BDM+V) consistently improves retrieval.

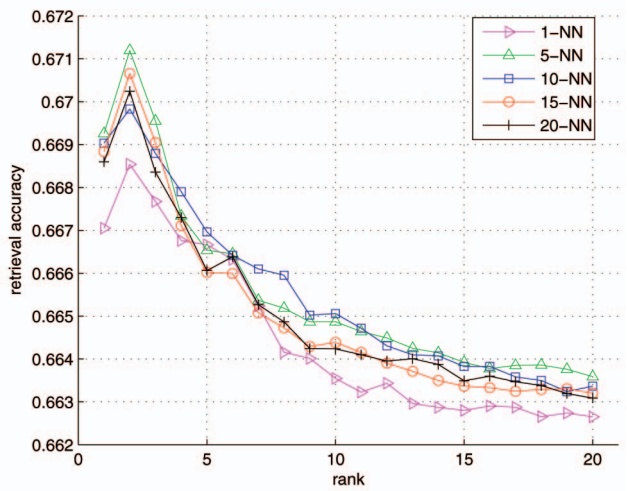


Fig. 6. Retrieval accuracy curves for BDM+V using different numbers of near neighbors to generate visual pairs. Retrieval is relatively insensitive to the number of visual pairs used in BDM+V.

retrieval accuracy for the euclidean distance. The standard deviation in the retrieval accuracy is illustrated by the vertical bar. First, we observe that the retrieval accuracy of both variants of BDM exceeds that of the euclidean distance metric, indicating that BDM is effective in learning appropriate distance functions. Second, we observe that the incorporation of visual pairs improves retrieval accuracy. This improvement can be explained from the viewpoint of Bayesian statistics since the visual pairs can be viewed as a Bayesian priors, as discussed above. Hence, BDM with visual pairs can be interpreted as Maximum A Posterior (MAP), while BDM without visual pairs can somehow be interpreted as Maximum-Likelihood Estimation (MLE). It is well known that MAP-based approaches typically outperform MLE-based approaches. This is particularly true when the number of training examples is not large in comparison to the number of parameters, allowing the target classification model to overfit the training examples. By introducing a Bayesian prior, we are able to regularize the fitting of the target classification model for the given training examples, thus alleviating the problem of overfitting.

In the second experiment, we evaluate the effect of the visual pairs on classification accuracy. We compute the area under the ROC curves (AUR), which is a common metric for evaluating classification accuracy. Table 1 shows the AUR results for BDM+V and BDM (i.e., with and without visual pairs) and the euclidean distance metric. Similarly to the previous experiment, we observe that areas under the ROC curves of the two variants of BDM are significantly larger than that of the euclidean distance, showing that BDM achieves better classification accuracy than the euclidean distance metric. Similarly to retrieval accuracy, we observe that the incorporation of visual pairs noticeably improves the classification accuracy.

The final experiment in this section is designed to study how different numbers of visual pairs affect the classification and retrieval performance. We vary the size of neighborhood from 1, 5, 10, and 15 to 20 when generating visual pairs. The larger the neighborhood size, the more visual pairs are generated. Fig. 6 and Table 2 show the retrieval accuracy and the area under ROC curves for BDM+V using different neighborhood sizes for generating visual pairs. We observe that the five different neighborhood sizes result in similar performance in both classification and retrieval. We thus conclude that BDM+V is overall insensitive to the number of visual pairs. Note that our study is limited to a modest range of visual pairs. The size of the euclidean near neighborhood should be controlled; otherwise, this approximation fails to capture visual similarity between images.

4.4 Comparison to State-of-the-Art Algorithms for Distance Metric Learning

We compare BDM+V to three state-of-the-art algorithms for learning distance functions and distance metrics: Linear

TABLE 1
Comparison of the Classification Accuracy

Algorithms	Area under ROC curve (AUR)
Euclidean	0.673 ± 0.004
BDM	0.722 ± 0.003
BDM+V	$\mathbf{0.736 \pm 0.003}$

The learned metrics result in better classification and the addition of visual pairs (BDM+V) is significant.

TABLE 2
Classification Results for BDM+V Using Different Numbers of Near Neighbors for Visual Pair Generation

Neighborhood size	Area under ROC curve
1	0.721 ± 0.003
5	0.722 ± 0.003
10	0.722 ± 0.003
15	0.721 ± 0.003
20	0.721 ± 0.004

BDM+V is relatively insensitive to the number of visual pairs used.

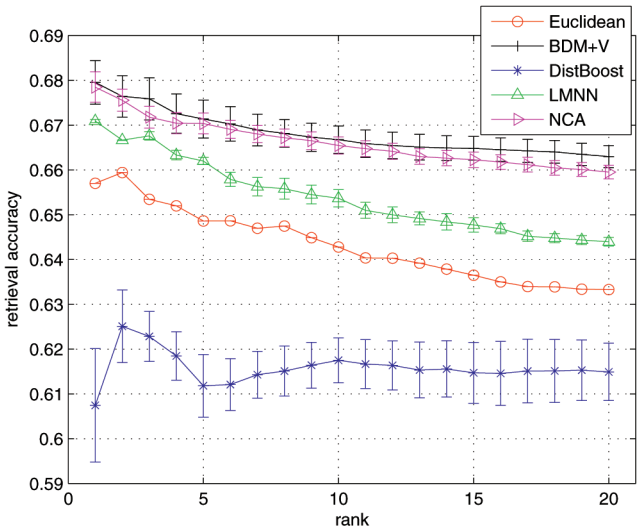


Fig. 7. Retrieval accuracy of distance metric learning algorithms on the mammogram data set.

Boost Distance (denoted as "DistBoost") [50], Large Margin Nearest Neighbor Classifier (denoted as "LMNN") [6], and Neighborhood Component Analysis (denoted as "NCA") [30]. Euclidean distance is included as a comparative reference (denoted as "euclidean").

4.4.1 Results on UPMC Mammograms Data Set

Fig. 7 shows the retrieval accuracy curves for BDM+V and the three comparative algorithms for distance metric learning. First, we observe that all of the distance learning algorithms outperform the euclidean distance metric except for the DistBoost algorithm which performs considerably worse than the euclidean distance metric. Second, BDM+V and NCA perform consistently better than the other algorithms across all the ranking positions. Table 3 shows the area under the ROC curve for BDM+V and the baseline methods. The proposed algorithm has the largest area under the ROC curve, followed by LMNN, euclidean, NCA, and finally, DistBoost. It is interesting to observe that although NCA achieves a better retrieval accuracy than the euclidean distance, its classification accuracy is considerably lower than the euclidean distance.

4.4.2 Results on the ImageCLEF Data Set

To generalize the performance of the proposed algorithm, we further evaluate the proposed algorithm on the medical

TABLE 3
Classification Accuracy on the Mammogram Data Set

Algorithms	Area under ROC curve
Euclidean	0.673 ± 0.004
DistBoost	0.601 ± 0.011
NCA	0.614 ± 0.004
LMNN	0.683 ± 0.004
BDM+V	$\mathbf{0.698 \pm 0.003}$

Fig. 8. Examples of medical images in the ImageCLEF testbed.

image data set provided by the ImageCLEF conference [60]. This is a popular benchmark data set used to evaluate automated medical image categorization and retrieval. It consists of 15 medical image categories with a total of 2,785 images. All of the medical images in this experiment are X-ray images collected from plain radiography. Fig. 8 shows a few examples of medical images in our testbed. The category information can be found from the conference Web site.

Following the typical practice in ImageCLEF, we process each medical image using a bank of Gabor wavelet filters [63] to extract texture features. More specifically, each image is first scaled to the size of 128×128. Then, the Gabor wavelet transform is applied to the scaled image at five scale levels and eight orientations, which results in a total of 40 subimages. Every subimage is further normalized into $8 \times 8 = 64$ features, which results in a total of $64 \times 40 = 2{,}560$ visual features for each medical image. PCA is used to reduce the dimensionality from 2,560 to 200. We select a total of 1,100 images from 11 categories in the ImageCLEF for our experiments. We randomly selected 40 percent images for the training data set and the remaining images serve as test queries.

The retrieval accuracy, defined in (20), is reported in Fig. 9. It is interesting to observe that NCA, which achieves high retrieval accuracy on the UPMC Mammogram Data Set, now performs significantly worse than the euclidean distance metric. On the other hand, DistBoost, which

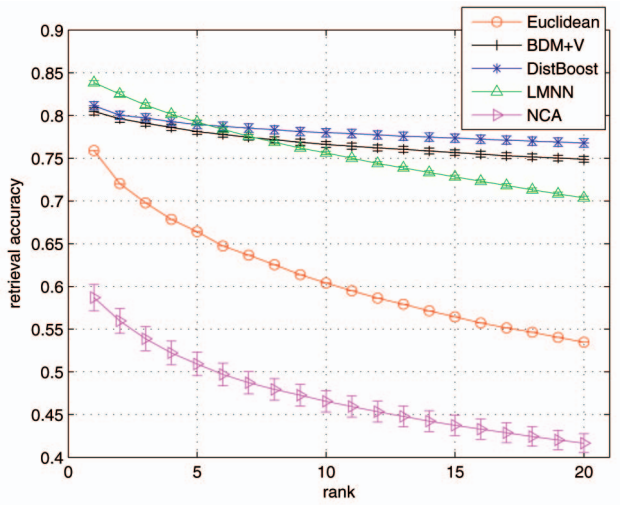


Fig. 9. Retrieval accuracy by different distance metric learning algorithms on the ImageCLEF data set.

TABLE 4
Area under ROC Curve on the ImageCLEF Data Set, Obtained by the Proposed Baseline Algorithms

Class	Euclidean	BDM+V	NCA	LMNN	DistBoost
1	0.973 ± 0.003	$\mathbf{0.998 \pm 0.001}$	0.995 ± 0.001	0.996 ± 0.001	0.990 ± 0.003
2	0.956 ± 0.004	0.988 ± 0.002	0.987 ± 0.002	$\mathbf{0.993 \pm 0.001}$	0.978 ± 0.003
3	0.911 ± 0.005	0.973 ± 0.003	0.925 ± 0.010	$\mathbf{0.987 \pm 0.001}$	0.972 ± 0.006
4	0.941 ± 0.009	0.986 ± 0.001	0.932 ± 0.004	$\mathbf{0.993 \pm 0.001}$	0.982 ± 0.003
5	0.862 ± 0.012	0.932 ± 0.002	0.780 ± 0.027	$\mathbf{0.942 \pm 0.005}$	0.928 ± 0.004
6	0.851 ± 0.007	0.960 ± 0.004	0.859 ± 0.014	$\mathbf{0.972 \pm 0.004}$	0.941 ± 0.009
7	0.788 ± 0.015	0.950 ± 0.005	0.826 ± 0.019	$\mathbf{0.963 \pm 0.005}$	0.958 ± 0.005
8	0.927 ± 0.004	0.985 ± 0.002	0.896 ± 0.012	$\mathbf{0.985 \pm 0.001}$	0.980 ± 0.003
9	0.825 ± 0.019	0.899 ± 0.008	0.753 ± 0.008	$\mathbf{0.922 \pm 0.006}$	0.882 ± 0.008
10	0.866 ± 0.015	0.988 ± 0.002	0.853 ± 0.017	$\mathbf{0.991 \pm 0.001}$	0.980 ± 0.005
11	0.881 ± 0.010	0.978 ± 0.003	0.854 ± 0.011	$\mathbf{0.983 \pm 0.003}$	0.963 ± 0.007

performed poorly on the UPMC data set, is one of the best algorithms. This result indicates that some of the state-of-the-art distance metric learning algorithms are sensitive to the characteristics of data sets and their performance is usually data-dependent. In contrast, BDM+V achieves good retrieval accuracy on both data sets, indicating the robustness of the proposed algorithm.

We also conduct the classification experiment using the ImageCLEF data set. Table 4 summarizes the area under the ROC curve for all the 11 classes separately. As we observe, for most classes, BDM+V achieves a performance that is comparable to LMNN, the best among the five competitors.

4.5 Computational Cost

As discussed in Section 1, high computational cost is one of the major challenges in learning distance metrics. Many approaches aim to learn a full matrix and therefore become computationally expensive as the dimensionality grows. BDM+V reduces the computational cost by learning a binary representation in a boosting framework, from which a weighted Hamming distance is computed. Table 5 shows the running time of the proposed algorithm and the baseline methods, for different dimensionality using the ImageCLEF data set. Note that the different numbers of dimensions are created by applying PCA to the images in the database and selecting the top eigenvectors for representing images.

First, the proposed algorithm is considerably faster than the three competitors when each image is represented by more than 200 features. Second, the time consumed by the proposed algorithm does not increase dramatically as the number of dimensions increases from 100 to 500; in contrast, for the three baseline algorithms, we observe a significant increase in the computational time as the dimension grows beyond 300. For instance, DistBoost is impressively fast (524.1 seconds) with 200 dimensions but falls behind BDM+V when the dimension increases to 300, and this gap widens in the case of 400 and 500 dimensions. NCA is the most computationally expensive among the four competitors, starting at 1,896.1 seconds for 100 dimensions and rising rapidly to end at 84,016.9 seconds for 500 dimensions. From these experiments, it is evident that, for all of the baseline methods, the efficiency issue becomes severe with higher dimensionality. In contrast, due to its efficient design, the computational time for the proposed method increases only linearly with respect to the dimensionality.

TABLE 5
Computation Time (Seconds) for the Proposed and Baseline Algorithms as the Number of Dimensions Varies from 100 to 500

# dim	100	200	300	400	500
BDM+V	568.9	673.1	**767.9**	**905.9**	**1087.6**
LMNN	**384.9**	913.0	3566.9	7628.9	12514.8
DistBoost	239.5	**524.1**	1154.5	2123.8	3627.2
NCA	1896.1	10744.3	33591.0	50384.9	84016.9

4.6 Regular Image Retrieval on the COREL Data Set

To demonstrate the efficacy of BDM+V for regular image retrieval, we test the proposed algorithm on the COREL data set. We randomly choose 10 categories from the COREL data set and randomly select 100 examples from each category, resulting in an image collection of 1,000 images. Each image is represented by 36 different visual features that belong to three categories: color, edge, and texture. The details of the visual feature used to represent the COREL data set can be found in [31].

The retrieval accuracy is reported in Fig. 10. Although the proposed algorithm **BDM+V** is outperformed overall by LMNN and DistBoost, we observe that **BDM+V** surpasses DistBoost at the first rank and outperforms LMNN after rank 14.

Table 6 reports the area under the ROC curve for all the 11 classes separately. **BDM+V** performs comparably to LMNN, which achieves the best results across the 10 classes. The other three competitors, i.e., DistBoost, NCA, and euclidean, often perform significantly worse than LMNN and the proposed algorithm. Moreover, the standard deviation of **BDM+V** and LMNN is, in general, smaller than the three baselines, indicating the robustness of the proposed algorithm.

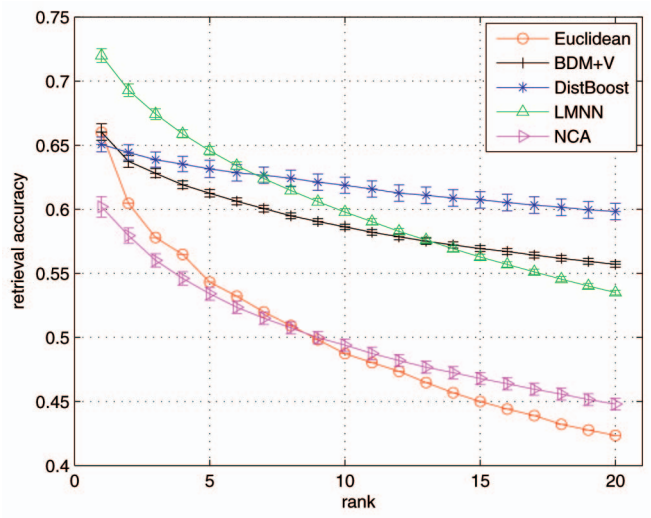


Fig. 10. Retrieval accuracy on the Corel data set.

5 Conclusion and Discussions

In this paper, we present a novel framework that learns a distance metric from side information. Unlike the other distance metric learning algorithms that are designed to learn a full matrix for distance metric, and therefore, suffer from computational difficulty, the proposed algorithm first learns a binary representation for data and then computes the weighted Hamming distance based on the learned representation. A boosting algorithm is presented to facilitate the learning of the binary representation and the weights that are used to form the Hamming distance. In addition to the computational efficiency, another advantage of the proposed algorithm is that it is able to preserve both the semantic relevance and the visual similarity. This is realized through the introduction of links that pair visually similar images. By training over the combination of visual pairs and pairwise constraints that are generated based on semantic relevance, the resulting distance metric is able to preserve both the visual similarity and semantical relevance. In contrast, the previous work on distance metric learning tends to focus only on the semantic relevance. We demonstrate the effectiveness of the proposed algorithm in the context of an ISADSs system for breast cancer and on two standard image data sets (ImageCLEF and Corel).

Appendix A

Proof of Lemma 3.2

To prove the inequality, we consider the following two cases:

- $f(\mathbf{x}_i) = f(\mathbf{x}_j)$: In this case, the inequality in (3) holds because the left side of the inequality is zeros and the right side is guaranteed to be nonnegative.
- $f(\mathbf{x}_i) \neq f(\mathbf{x}_j)$: In this case, $f(\mathbf{x}_k)$ will be equal to either $f(\mathbf{x}_i)$ or $f(\mathbf{x}_j)$ since $f(\mathbf{x})$ is a binary function. Hence, both sides of the inequality are equal to 4, and therefore, the inequality in (3) holds.

Appendix B

Proof of Lemma 3.3

To prove the inequality in (11), we first bound $\exp(\alpha((f_i - f_k)^2 - (f_i - f_j)^2))$ by the following expression:

$$\exp(\alpha((f_i - f_k)^2 - (f_i - f_j)^2)) \leq \frac{\exp(2\alpha(f_i - f_k)^2)}{2} + \frac{\exp(-2\alpha(f_i - f_j)^2)}{2}.$$

Since $f_i^2 = 1$ for any example $\mathbf{x}_i$, we have

$$\frac{(f_i - f_j)^2}{4} + \frac{(f_i + f_j)^2}{4} = 1.$$

Hence, $\exp(2\alpha(f_i - f_j)^2)$ can be upper bounded as follows:

$$\begin{aligned}
&\exp(2\alpha(f_i - f_j)^2) \\
&= \exp\left(8\alpha\frac{(f_i - f_j)^2}{4} + 0 \times \frac{(f_i + f_j)^2}{4}\right) \\
&\leq \frac{(f_i - f_j)^2}{4}\exp(8\alpha) + \frac{(f_i + f_j)^2}{4} \\
&= \frac{(f_i - f_j)^2}{4}(\exp(8\alpha) - 1) + 1.
\end{aligned}$$

TABLE 6
Area under the ROC Curve on the Corel Data Set, Obtained by the Proposed and Baseline Algorithms

Class	Euclidean	BDM+V	NCA	LMNN	DistBoost
1	0.735 ± 0.017	0.841 ± 0.006	0.839 ± 0.006	$\mathbf{0.881 \pm 0.005}$	0.814 ± 0.010
2	0.849 ± 0.008	0.965 ± 0.004	0.965 ± 0.006	$\mathbf{0.981 \pm 0.003}$	0.964 ± 0.005
3	0.848 ± 0.006	0.950 ± 0.004	0.951 ± 0.003	$\mathbf{0.975 \pm 0.002}$	0.961 ± 0.002
4	0.900 ± 0.004	0.978 ± 0.002	0.947 ± 0.004	$\mathbf{0.991 \pm 0.001}$	0.968 ± 0.005
5	0.881 ± 0.006	0.930 ± 0.004	0.918 ± 0.006	$\mathbf{0.949 \pm 0.003}$	0.918 ± 0.004
6	0.750 ± 0.009	0.895 ± 0.004	0.850 ± 0.006	$\mathbf{0.920 \pm 0.006}$	0.905 ± 0.004
7	0.921 ± 0.005	0.952 ± 0.005	0.913 ± 0.006	$\mathbf{0.960 \pm 0.006}$	0.947 ± 0.005
8	0.758 ± 0.005	0.822 ± 0.003	0.816 ± 0.007	$\mathbf{0.882 \pm 0.006}$	0.863 ± 0.008
9	0.823 ± 0.008	0.964 ± 0.003	0.946 ± 0.004	$\mathbf{0.970 \pm 0.005}$	0.960 ± 0.004
10	0.771 ± 0.008	0.815 ± 0.010	0.751 ± 0.017	0.862 ± 0.005	$\mathbf{0.867 \pm 0.008}$

Using the above inequality, we have the objective function $F(\mathcal{P})$ in (9) upper bounded as follows:

$$\begin{aligned}
&F(\mathcal{P}) - \sum_{i,j,k=1}^{n} \phi_{i,j}^{-}\phi_{i,k}^{+} \\
&= \sum_{i,j,k=1}^{n} \phi_{i,j}^{-}\phi_{i,k}(\exp(\alpha(f_i - f_j)^2 - \alpha(f_i - f_k)^2) - 1) \\
&\leq \frac{\exp(-8\alpha) - 1}{8} \sum_{i,j=1}^{n} \phi_{i,j}^{-} \left(\sum_{k=1}^{n} \phi_{i,k}^{+} \right) (f_i - f_j)^2 \\
&\quad + \frac{\exp(8\alpha) - 1}{8} \sum_{i,j=1}^{n} \phi_{i,j}^{+} \left(\sum_{k=1}^{n} \phi_{i,k}^{-} \right) (f_i - f_j)^2 \\
&= \frac{\exp(-8\alpha) - 1}{8} \mathbf{f}^\top L^{+} \mathbf{f} + \frac{\exp(8\alpha) - 1}{8} \mathbf{f}^\top L^{-} \mathbf{f}.
\end{aligned}$$

The last step of the above derivation is based on the following equality:

$$\mathbf{f}^\top L(S) \mathbf{f} = \sum_{i,j=1}^{n} S_{i,j}(f_i - f_j)^2.$$

Finally, noting that $\tilde{F}(\mathcal{P})$, i.e., the objective function of previous iteration, is equal to $\sum_{i,j,k=1}^{n} \phi_{i,k}^{-}\phi_{i,j}^{+}$, we have $\Delta(\alpha, \mathbf{f}) = F(\mathcal{P}) - \tilde{F}(\mathcal{P})$ upper bounded as follows:

$$\Delta(\alpha, \mathbf{f}) \leq \frac{\exp(-8\alpha) - 1}{8} \mathbf{f}^\top L^{+} \mathbf{f} + \frac{\exp(8\alpha) - 1}{8} \mathbf{f}^\top L^{-} \mathbf{f}.$$

APPENDIX C
PROOF OF THEOREM 3.4

We first denote by $g(\alpha, \mathbf{f})$ the right-hand side of the inequality in (11), i.e.,

$$g(\alpha, \mathbf{f}) = \frac{\exp(-8\alpha) - 1}{8} \mathbf{f}^\top L^{+} \mathbf{f} + \frac{\exp(8\alpha) - 1}{8} \mathbf{f}^\top L^{-} \mathbf{f}.$$

Note that $g(\alpha, \mathbf{f})$ is a convex function of parameter α. We then compute $\min_{\alpha \geq 0} g(\alpha, \mathbf{f})$ by setting the first order derivative of α to be zero, i.e.,

$$\frac{\partial g(\alpha, \mathbf{f})}{\partial \alpha} = -\exp(-8\alpha) \mathbf{f}^\top L^{+} \mathbf{f} + \exp(8\alpha) \mathbf{f}^\top L^{-} \mathbf{f} = 0.$$

We obtain the optimal α by solving the above equation, which is

$$\alpha = \max\left(0, \frac{1}{16} \log\left(\mathbf{f}^\top L^{+} \mathbf{f}\right) - \frac{1}{16} \log\left(\mathbf{f}^\top L^{-} \mathbf{f}\right)\right).$$

Substituting the above expression for α, we have

$$\begin{aligned}
\min_{\alpha \geq 0} g(\alpha, \mathbf{f}) &\leq -\frac{1}{8} \left(\max\left(0, \sqrt{\mathbf{f}^\top L^{+} \mathbf{f}} - \sqrt{\mathbf{f}^\top L^{-} \mathbf{f}}\right)\right)^2 \\
&= -\frac{\left(\max(0, \mathbf{f}^\top L^{+} \mathbf{f} - \mathbf{f}^\top L^{-} \mathbf{f})\right)^2}{8(\sqrt{\mathbf{f}^\top L^{+} \mathbf{f}} + \sqrt{\mathbf{f}^\top L^{-} \mathbf{f}})^2} \\
&\leq -\frac{\left(\max(0, \mathbf{f}^\top L^{+} \mathbf{f} - \mathbf{f}^\top L^{-} \mathbf{f})\right)^2}{8n(\sqrt{\lambda_{\max}(L^{-})} + \sqrt{\lambda_{\max}(L^{+})})^2}.
\end{aligned}$$

Since $\Delta(\alpha, \mathbf{f}) \leq g(\alpha, \mathbf{f})$, we have the bound in (15).

APPENDIX D
PROOF OF THEOREM 3.5

According to Theorem 3.4, we have

$$\frac{F_{t+1}(\mathcal{P})}{F_t(\mathcal{P})} \leq 1 - \frac{[\max(0, \mathbf{f}^\top (L^{+} - L^{-}) \mathbf{f})]^2}{8nF_t(\mathcal{P})(\sqrt{\lambda_{\max}(L_t^{-})} + \sqrt{\lambda_{\max}(L_t^{+})})^2}. \quad (21)$$

Since we choose $\mathbf{f}$ to maximize the $\mathbf{f}(L_t^{+} - L_t^{-})\mathbf{f}$, we have

$$\max(0, \max_{\mathbf{f}} \mathbf{f}^\top (L_t^{+} - L_t^{-}) \mathbf{f}) = \lambda_{\max}(L_t^{+} - L_t^{-}). \quad (22)$$

The above derivation uses the following fact:

$$\lambda_{\max}(L_t^{+} - L_t^{-}) \geq \frac{1}{n} (\mathbf{1}^\top (L_t^{+} - L_t^{-}) \mathbf{1}) = 0.$$

We can further simplify the bound in (21) by having

$$\left(\sqrt{\lambda_{\max}(L_t^{-})} + \sqrt{\lambda_{\max}(L_t^{+})}\right)^2 \leq 2(\lambda_{\max}(L_t^{-}) + \lambda_{\max}(L_t^{+})). \quad (23)$$

Finally, we can upper bound $F_t(\mathcal{P})$ as follows:

$$F_t(\mathcal{P}) = \sum_{i,j,k=1}^{n} \phi_{i,j}\phi_{i,k} = \frac{1}{2} \mathbf{1}^\top (S_t^{+} + S_t^{-}) \mathbf{1} \leq \frac{1}{2} \lambda_{\max}(S_t^{+} + S_t^{-}). \quad (24)$$

By putting the inequalities in (22), (23), and (24), we have

$$\begin{aligned}
\frac{F_{t+1}(\mathcal{P})}{F_t(\mathcal{P})} &\leq 1 - \frac{[\lambda_{\max}(L_t^{+} - L_t^{-})]^2}{8(\lambda_{\max}(S_t^{+} + S_t^{-}))(\lambda_{\max}(L_t^{+}) + \lambda_{\max}(L_t^{-}))} \\
&= 1 - \gamma_t.
\end{aligned}$$

Using the above inequality, we can bound $F_{T+1}(\mathcal{P})$ as follows:

$$F_{T+1}(\mathcal{P}) = F_0(\mathcal{P}) \prod_{t=0}^{T} \frac{F_{t+1}(\mathcal{P})}{F_t(\mathcal{P})} \leq F_0(\mathcal{P}) \prod_{t=0}^{T} (1 - \gamma_t).$$

ACKNOWLEDGMENTS

This work was supported by the US National Science Foundation (NSF) under grant IIS-0643494 and by the National Center for Research Resources (NCRRs) under grant No. 1 UL1 RR024153. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF, NCRR, Intel, Michigan State University, or Carnegie Mellon University.

REFERENCES

[1] P. Croskerry, "The Theory and Practice of Clinical Decision-Making," *Canadian J. Anesthesia*, vol. 52, no. 6, pp. R1-R8, 2005.
[2] L. Yang, R. Jin, R. Sukthankar, B. Zheng, L. Mummert, M. Satyanarayanan, M. Chen, and D. Jukic, "Learning Distance Metrics for Interactive Search-Assisted Diagnosis of Mammograms," *Proc. SPIE Conf. Medical Imaging*, 2007.
[3] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-Based Image Retrieval at the End of the Early Years," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 22, no. 12, pp. 1349-1380, Dec. 2000.

[4] H. Kim, P. Howland, and H. Park, "Dimension Reduction in Text Classification with Support Vector Machines," *J. Machine Learning Research,* vol. 6, pp. 37-53, 2005.

[5] L. Vandenberghe and S. Boyd, "Semidefinite Programming," *SIAM Rev.,* vol. 38, no. 1, pp. 49-95, 1996.

[6] K. Weinberger, J. Blitzer, and L. Saul, "Distance Metric Learning for Large Margin Nearest Neighbor Classification," *Advances in Neural Information Processing Systems,* MIT Press, http://www.seas.upenn.edu/kilianw/lmnn, 2006.

[7] D. Gur, J.H. Sumkin, L.A. Hardesty, and H.E. Rockette, "Computer-Aided Detection of Breast Cancer: Has Promise Outstripped Performance?" *J. Nat'l Cancer Inst.,* vol. 96, pp. 717-718, 2004.

[8] R.M. Nishikawa and M. Kallergi, "Computer-Aided Detection in Its Present Form Is Not an Effective Aid for Screening Mammography," *Medical Physics,* vol. 33, pp. 811-814, 2006.

[9] T.M. Freer and M.J. Ulissey, "Screening Mammography with Computer-Aided Detection: Prospective Study of 12,860 Patients in a Community Breast Center," *Radiology,* vol. 220, pp. 781-786, 2001.

[10] L.A. Khoo, P. Taylor, and R.M. Given-Wilson, "Computer-Aided Detection in the United Kingdom National Breast Screening Programme: Prospective Study," *Radiology,* vol. 237, pp. 444-449, 2005.

[11] J.M. Ko, M.J. Nicholas, J.B. Mendel, and P.J. Slanetz, "Prospective Assessment of Computer-Aided Detection in Interpretation of Screening Mammograms," *Am. J. Roentgenology,* vol. 187, pp. 1483-1491, 2006.

[12] M.L. Giger, Z. Huo, C.J. Vyborny, L. Lam, I. Bonta, K. Horsch, R.M. Nishikawa, and I. Rosenbourgh, "Intelligent CAD, Workstation for Breast Imaging Using Similarity to Known Lesions and Multiple Visual Prompt Aides," *Proc. SPIE Conf. Medical Imaging '02: Image Processing,* pp. 768-773, 2002.

[13] I. El-Naga, Y. Yang, N.P. Galatsanos, R.M. Nishikawa, and M.N. Wernick, "A Similarity Learning Approach to Content-Based Image Retrieval: Application to Digital Mammography," *IEEE Trans. Medical Imaging,* vol. 23, no. 10, pp. 1233-1244, Oct. 2004.

[14] C. Wei, C. Li, and R. Wilson, "A General Framework for Content-Based Medical Image Retrival with Its Application to Mammograms," *Proc. SPIE Conf. Medical Imaging '05: PACS and Imaging Informatics,* pp. 134-143, 2005.

[15] H. Alto, R.M. Rangayyan, and J.E. Desautels, "Content-Based Retrieval and Analysis of Mammographic Masses," *J. Electronic Imaging,* vol. 14, pp. 023016-1-023016-17, 2005.

[16] C. Muramatsu, Q. Li, K. Suzuki, R.A. Schmidt, J. Shiraishi, G.M. Newstead, and K. Doi, "Investigation of Psychophysical Measure for Evaluation of Similar Images for Mammographic Masses: Preliminary Results," *Medical Physics,* vol. 32, pp. 2295-2304, 2005.

[17] B. Zheng et al., "Interactive Computer Aided Diagnosis of Breast Masses: Computerized Selection of Visually Similar Image Sets from a Reference Library," *Academic Radiology,* vol. 14, no. 8, pp. 917-927, 2007.

[18] G.D. Tourassi, B. Harrawood, S. Singh, J.Y. Lo, and C.E. Floyd, "Evaluation of Information-Theoretic Similarity Measures for Content-Based Retrieval and Detection of Masses in Mammograms," *Medical Physics,* vol. 34, pp. 140-150, 2007.

[19] Y. Tao, S.B. Lo, M.T. Freedman, and J. Xuan, "A Preliminary Study of Content-Based Mammographic Masses Retrieval Book," *Proc. SPIE Conf. Medical Imging '07,* 2007.

[20] R.M. Nishikawa, "Current Status and Future Directions of Computer-Aided Diagnosis in Mammography," *Computerized Medical Imaging Graphics,* vol. 31, pp. 224-235, 2007.

[21] L. Yang and R. Jin, "Distance Metric Learning: A Comprehensive Survey," technical report, Michigan State Univ., 2006.

[22] E. Xing, A. Ng, M. Jordan, and S. Russell, "Distance Metric Learning with Application to Clustering with Side Information," *Advances in Neural Information Processing Systems,* MIT Press, 2003.

[23] J.T. Kwok and I.W. Tsang, "Learning with Idealized Kernels," *Proc. Int'l Conf. Machine Learning,* 2003.

[24] T. Hastie and R. Tibshirani, "Discriminant Adaptive Nearest Neighbor Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, no. 6, pp. 607-616, June 1996.

[25] A.B. Hillel, T. Hertz, N. Shental, and D. Weinshall, "Learning Distance Functions Using Equivalence Relations," *Proc. Int'l Conf. Machine Learning,* 2003.

[26] S.C.H. Hoi, W. Liu, M.R. Lyu, and W.-Y. Ma, "Learning Distance Metrics with Contextual Constraints for Image Retrieval," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2006.

[27] M. Sugiyama, "Local Fisher Discriminant Analysis for Supervised Dimensionality Reduction," *Proc. Int'l Conf. Machine Learning,* 2006.

[28] T.-K. Kim, S.-F. Wong, B. Stenger, J. Kittler, and R. Cipolla, "Incremental Linear Discriminant Analysis Using Sufficient Spanning Set Approximations," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2007.

[29] M. Schultz and T. Joachims, "Learning a Distance Metric from Relative Comparisons," *Advances in Neural Information Processing Systems,* MIT Press, 2004.

[30] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood Components Analysis," *Advances in Neural Information Processing Systems,* MIT Press, 2005.

[31] L. Yang, R. Jin, R. Sukthankar, and Y. Liu, "An Efficient Algorithm for Local Distance Metric Learning," *Proc. Nat'l Conf. Artificial Intelligence,* 2006.

[32] A.B. Hillel and D. Weinshall, "Learning Distance Function by Coding Similarity," *Proc. Int'l Conf. Machine Learning,* 2007.

[33] A. Woznica, A. Kalousis, and M. Hilario, "Learning to Combine Distances for Complex Representations," *Proc. Int'l Conf. Machine Learning,* 2007.

[34] W. Zhang, X. Xue, Z. Sun, Y. Guo, and H. Lu, "Optimal Dimensionality of Metric Space for Classification," *Proc. Int'l Conf. Machine Learning,* 2007.

[35] H. Wang, H. Zha, and H. Qin, "Dirichlet Aggregation: Unsupervised Learning Towards an Optimal Metric for Proportional Data," *Proc. Int'l Conf. Machine Learning,* 2007.

[36] S. Zhou, B. Georgescu, D. Comaniciu, and J. Shao, "BoostMotion: Boosting a Discriminative Similarity Function for Motion Estimation," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2006.

[37] B. Babenko, P. Dollár, and S. Belongie, "Task Specific Local Region Matching," *Proc. Int'l Conf. Computer Vision,* 2007.

[38] F. Li, J. Yang, and J. Wang, "A Transductive Framework of Distance Metric Learning by Spectral Dimensionality Reduction," *Proc. Int'l Conf. Machine Learning,* 2007.

[39] P. Dollár, V. Rabaud, and S. Belongie, "Non-Isometric Manifold Learning: Analysis and an Algorithm," *Proc. Int'l Conf. Machine Learning,* 2007.

[40] J.V. Davis, B. Kulis, P. Jain, S. Sra, and I.S. Dhillon, "Information-Theoretic Metric Learning," *Proc. Int'l Conf. Machine Learning,* 2007.

[41] J. Dillon, Y. Mao, G. Lebanon, and J. Zhang, "Statistical Translation, Heat Kernels, and Expected Distance," *Proc. Conf. Uncertainty in Artificial Intelligence,* 2007.

[42] L. Yang, R. Jin, and R. Sukthankar, "Bayesian Active Distance Metric Learning," *Proc. Conf. Uncertainty in Artificial Intelligence,* 2007.

[43] L. Torresani and K. Lee, "Large Margin Component Analysis," *Advances in Neural Information Processing Systems,* MIT Press, 2007.

[44] K.Q. Weinberger, F. Sha, Q. Zhu, and L.K. Saul, "Graph Laplacian Regularization for Large-Scale Semidefinite Programming," *Advances in Neural Information Processing Systems,* MIT Press, 2007.

[45] D. Zhou, J. Huang, and B. Schölkopf, "Learning with Hypergraphs: Clustering, Classification, and Embedding," *Advances in Neural Information Processing Systems,* MIT Press, 2007.

[46] Z. Zhang and J. Wang, "MLLE: Modified Locally Linear Embedding Using Multiple Weights," *Advances in Neural Information Processing Systems,* MIT Press, 2007.

[47] A. Frome, Y. Singer, and J. Malik, "Image Retrieval and Classification Using Local Distance Functions," *Advances in Neural Information Processing Systems,* MIT Press, 2007.

[48] O. Boiman and M. Irani, "Similarity by Composition," *Advances in Neural Information Processing Systems,* MIT Press, 2007.

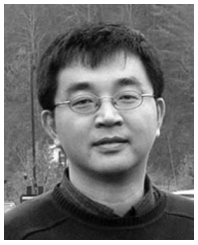
[49] M. Belkin and P. Niyogi, "Convergence of Laplacian Eigenmaps," *Advances in Neural Information Processing Systems,* MIT Press, 2007.

[50] T. Hertz, A.B. Hillel, and D. Weinshall, "Boosting Margin Based Distance Functions for Clustering," *Proc. Int'l Conf. Machine Learning,* http://www.cs.huji.ac.il/daphna/code/DistBoost.zip, 2004.

[51] T. Hertz, A.B. Hillel, and D. Weinshall, "Learning a Kernel Function for Classification with Small Training Samples," *Proc. Int'l Conf. Machine Learning,* 2006.

[52] G. Shakhnarovich, "Learning Task-Specific Similarity," PhD thesis, Massachusetts Inst. of Technology, 2005.
[53] Y. Freund and R.E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *J. Computer and System Sciences,* vol. 55, no. 1, pp. 119-139, 1997.
[54] S. Agarwal, J. Wills, L. Cayton, G. Lanckriet, D. Kriegman, and S. Belongie, "Generalized Non-Metric Multidimensional Scaling," *Proc. Int'l Conf. Artificial Intelligence and Statistics,* 2007.
[55] B. Moghaddham and M.-H. Yang, "Gender Classification with Support Vector Machines," *Proc. Int'l Conf. Face and Gesture Recognition,* 2000.
[56] Y. Ke, D. Hoiem, and R. Sukthankar, "Computer Vision for Music Identification," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2005.
[57] J. Zhang and R. Yan, "On the Value of Pairwise Constraints in Classification and Consistency," *Proc. Int'l Conf. Machine Learning,* 2007.
[58] R.E. Schapire, "Theoretical Views of Boosting and Applications," *Proc. Int'l Conf. Algorithmic Learning Theory,* 1999.
[59] R. Salakhutdinov, S. Roweis, and Z. Ghahramani, "On the Convergence of Bound Optimization Algorithms," *Proc. Conf. Uncertainty in Artificial Intelligence,* 2003.
[60] ImageCLEF, http://ir.shef.ac.uk/imageclef/, 2009.
[61] B. Zheng, J.K. Leader, G. Abrams, B. Shindel, V. Catullo, W.F. Good, and D. Gur, "Computer-Aided Detection Schemes: The Effect of Limiting the Number of Cued Regions in Each Case," *Am. J. Roentgenology,* vol. 182, pp. 579-583, 2004.
[62] B. Zheng, A. Lu, L.A. Hardesty, J.H. Sumkin, C.M. Kakim, M.A. Ganott, and D. Gur, "A Method to Improve Visual Similarity of Breast Masses for an Interactive Computer-Aided Diagnosis Environment," *Medical Physics,* vol. 33, pp. 111-117, 2006.
[63] M. Lades, J.C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. Wurtz, and W. Konen, "Distortion Invariant Object Trcognition in the Dynamic Line Architecture," *IEEE Trans. Computers,* vol. 42, no. 3, pp. 300-311, Mar. 1993.

Liu Yang received the BS degree in electronics and information engineering from Hua Zhong University of Science and Technology, China. She is currently working toward the PhD degree in the Machine Learning Department of the School of Computer Science at Carnegie Mellon University. Her research interest is primarily on semi-supervised learning, distance metric earning, information retrieval, and object Recognition. She was selected as the machine learning department nominee from CMU for the IBM Fellowship. She is a student member of the IEEE.



Rong Jin received the BA degree in engineering from Tianjin University, the MS degree in physics from Beijing University, and the MS and PhD degrees in computer science from Carnegie Mellon University. He has been an associate professor in the Computer Science and Engineering Department at Michigan State University since 2008. His research is focused on statistical machine learning and its application to large-scale information management. He has published more than 80 conference and journal articles on the related topics. He received the US National Science Foundation (NSF) Career Award in 2006.

Lily Mummert received the PhD degree in computer science from Carnegie Mellon University in 1996. She is a research scientist at Intel Research Pittsburgh, working in the area of distributed systems. Before joining Intel in 2006, she was a research staff member at the IBM T.J. Watson Research Center, where she worked on problems in enterprise system management and contributed to several products. Her current research is focused on enabling interactive applications that process data from heterogeneous, potentially high-data-rate sensors such as video and audio. She is a member of the IEEE.

Rahul Sukthankar received the PhD degree in robotics from Carnegie Mellon University and the BSE degree in computer science from Princeton. He is a senior principal research scientist at Intel Research Pittsburgh and an adjunct research professor in the Robotics Institute at Carnegie Mellon. He was previously a senior researcher at HP/Compaq's Cambridge Research Lab and a research scientist at Just Research. His current research focuses on computer vision and machine learning, particularly in the areas of object recognition and information retrieval in medical imaging. He is a member of the IEEE.

Adam Goode received the bachelor's degree in computer science and psychology from Rensselaer Polytechnic Institute and the master's degree from the Human-Computer Interaction Institute at Carnegie Mellon University. He is a project scientist working at Carnegie Mellon, working on Diamond, a system for interactive search. He has been working as a research staff member at Carnegie Mellon since 2001. He is a member of the IEEE.

Bin Zheng received the PhD degree in electrical engineering from the University of Delaware. Currently, he is a research associate professor in the Department of Radiology at the University of Pittsburgh. He is also the principal investigator on a number of biomedical imaging research projects funded by the US National Institutes of Health (NIH). His research projects and interests include computer-aided detection and diagnosis (CAD) of medical images, content-based image retrieval, machine learning, and receiver operating characteristic (ROC)-type observer performance studies and data analysis. He and his colleagues have published more than 60 refereed articles in developing and evaluating CAD schemes and systems for digitized mammograms, lung CT images, and digital microscopic pathology images.

Steven C.H. Hoi received the BS degree in computer science from Tsinghua University, Beijing, China, and the MS and PhD degrees in computer science and engineering from the Chinese University of Hong Kong. He is currently an assistant professor in the School of Computer Engineering of Nanyang Technological University, Singapore. His research interests include statistical machine learning, multimedia information retrieval, Web search, and data mining. He is a member of the IEEE.

Mahadev Satyanarayanan received the bachelor's and master's degrees from the Indian Institute of Technology, Madras, and the PhD degree in computer science from Carnegie Mellon. He is the Carnegie Group professor of computer science at Carnegie Mellon University. From May 2001 to May 2004, he served as the founding director of Intel Research Pittsburgh, one of four university-affiliated research labs established worldwide by Intel to create disruptive information technologies through its Open Collaborative Research model. He is a fellow of the ACM and the IEEE and was the founding editor-in-chief of *IEEE Pervasive Computing*.