Learning Effective Image Metrics from Few Pairwise Examples

Hwann-Tzong Chen[1,2] Tyng-Luh Liu[1] Chiou-Shann Fuh[2]
[1]Institute of Information Science, Academia Sinica, Taipei 115, Taiwan
[2]Dept. of CSIE, National Taiwan University, Taipei 106, Taiwan

Abstract

We present a new approach to learning image metrics. The main advantage of our method lies in a formulation that requires only a few pairwise examples. Apparently, based on the little amount of side-information, it would take a very effective learning scheme to yield a useful image metric. Our algorithm achieves this goal by addressing two key issues. First, we establish a global-local (glocal) image representation that induces two structure-meaningful vector spaces to respectively describe the global and the local image properties. Second, we develop a metric optimization framework that finds an optimal bilinear transform to best explain the given side-information. We emphasize it is the glocal image representation that makes the use of bilinear transform more powerful. Experimental results on classifications of face images and visual tracking are included to demonstrate the contributions of the proposed method.

1. Introduction

The need of comparing two images is ubiquitous in many computer vision problems. Naturally, its effectiveness depends on the accuracy of the underlying similarity measure. Take, for example, the task of recognizing faces: a learned similarity measure for the face images may greatly improve the recognition rate, even with a simple classifier that applies the nearest neighbor rule. In addition, for some problems such as data clustering or image retrieval, one would prefer the similarity measure (or the distance function) to have metric properties. While designing a universal metric suitable for all images is too much to ask, we instead consider a practical but challenging problem of learning the task-dependent metrics, with the guidelines from a handful of user-specified examples. Fig. 1 illustrates a typical situation that using the Euclidean distance might not succeed in identifying the same person under different lighting conditions; however, it is probably good enough if we merely want to detect the lighting changes.

Our approach to image-metric learning can be characterized by four hallmarks: 1) The formulation requires only a few pairwise examples for the learning. In general, it would be time-consuming and rarely adequate to first analyze a large set of training data, and then re-analyze the whole set when new data are added. 2) The examples used in learning an image metric can be provided without specifying their labels. Instead, only side-information reflecting the relevance between pairs of images is considered [18], [20]. Indeed, it is more constructive to explore the relational information rather than the labels, when one could observe only a small number of samples. 3) A *glocal* image representation is introduced as the cornerstone of our proposed image-metric learning scheme. The representation accounts for both global and local intrinsic features of an image, and can be efficiently computed. As we shall explain later, this new image representation has several desirable properties over the conventional pixel-based vector representation. 4) With the glocal representation, an effective metric-learning algorithm can be derived by optimization through bilinear transforms/filterings. And the resulting image metric would describe more faithfully the specified relations between images, and operates in a lower dimensional space, a pivotal factor in speeding up the nearest-neighbor search.

Figure 1: The Euclidean distance cannot properly correlate the two images of the same person under different lighting conditions. For recognition, it should be preferable to have a distance function that describes the two images on the left to be closer than the two on the right.

Related Work. Learning distance metrics is crucial for various vision applications, e.g., object recognition [11], image retrieval [1], [8], [16], and video retrieval [9]. Most of these methods do not require the exact labels being provided with the training data, but work on the side-information [20] of the similarity relations. Typically, the side-information is given in the form of pairwise constraints that each prescribes a pair of data samples as similar (of the same class) or dissimilar (of different classes). Based on

a learned metric, the distance between every two samples can better quantify their class relations so that the accuracy of the nearest neighbor classification can be improved. An extreme case, different from learning with pairwise data, is to learn to classify objects from a single sample image [2], [4]. However, these approaches mostly rely on more complex feature descriptors. Besides improving classification accuracy, another aspect of consideration for metric learning is to increase the efficiency of nearest neighbor search. The *BoostMap* [1] is one example of such techniques that learns to embed the data into a low-dimensional Euclidean space, according to the similarity relations obtained from other computationally expensive similarity measures.

In fact methods on feature selection can also be viewed as metric-learning algorithms [6], [15], [19], since the goal is to give different weights on different dimensions. Let $\mathbf{w}$ contain the feature weights. Then the diagonal matrix D with elements $d_{ii} = w_i^2$ would yield the *Mahalanobis distance* between, say, $\mathbf{x}$ and $\mathbf{y}$ as $\sqrt{(\mathbf{x}-\mathbf{y})^T D(\mathbf{x}-\mathbf{y})}$. Gilad-Bachrach et al. [6] incorporate the margin defined for the nearest neighbor classification into the evaluation function of feature weights. Maximizing the evaluation function is analogous to learning the Mahalanobis distance that pulls each data point's nearest friend (with the same label) closer, and meanwhile, pushes away the nearest enemy (with a different label). More generally, a Mahalanobis distance can include a full matrix that is positive semi-definite. In this case the metric learning is related to the feature extraction problems, which are aimed to transform the data to another space, or to project the data onto a subspace that better represents the specific data relations [7], [16], [20].

2. Glocal Image Representations

Our formulation for learning an effective image metric starts with a novel matrix representation that economically encodes both the *global* correspondences and the *local* neighbors of each pixel. We call such an image descriptor *the glocal image representation* for convenience.

Consider now an image $\mathcal{I}_A$ of size $M \times N$ pixels. To derive its glocal (neighborhood) matrix A, we first partition $\mathcal{I}_A$ into n square blocks. In particular, using a block-size of $h \times h$ would give $n = \lfloor M/h \rfloor \times \lfloor N/h \rfloor$ image blocks (i.e., the remaining boundary pixels are ignored). Then each column of A can be obtained by raster-scanning an image block into a vector of length d ($= h \times h$). The resulting glocal matrix A is therefore of size $d \times n$. Note that the processing of image blocks also follows the same raster-scan order (see Fig. 2). Indeed, depending on the sequences of scanning the blocks and then the pixels in each block, there are four possible combinations to generate a glocal neighborhood matrix. We choose to scan by row for both the blocks and the pixels in each block. Nevertheless, it is

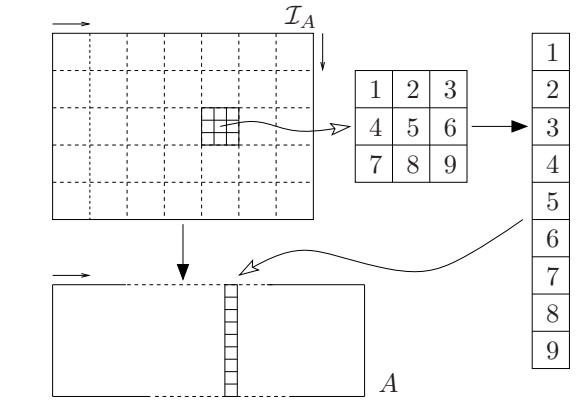

Figure 2: Deriving the glocal neighborhood matrix A from the image $\mathcal{I}_A$. In this example, the block size is 3×3 ($h = 3$), and A is of size 9×35.

easy to check that the glocal matrices derived from different scanning orders can be transformed into one another by row or column permutations. That is, they are equivalent up to a multiplication of some suitable permutation matrix.

The column space of a glocal matrix is spanned by vectors that depict the local image features (within each block), while the row space is generated from those that globally sample image features (from all blocks) of the original image (see Fig. 3). Thus the intrinsic local and global properties of an image are arranged in a way that techniques on matrix analysis can be conveniently applied to explore both aspects. Compared with using other filter-based image features, constructing the glocal matrix of an input image is much faster. Furthermore, since d and n are both smaller than $M \times N$, to solve the underlying optimization problems for metric learning based on the glocal representation is much more efficient and stable, especially when dealing with a small number of examples (the well-known *curse of dimensionality*). For instance, the typical image size in our experiments is $33 \times 33 = 1089$. We have $d = 9$ ($h = 3$), and $n = 11 \times 11 = 121$. The number of pairwise examples used can be as small as 10. Clearly, it would be more difficult to model the data in a space of 1089 dimensions than in a space of only 9 or 121 dimensions.

3. Learning Image Metrics

On learning task-dependent image metrics, we emphasize the case of using a few pairwise examples as the training data. The consideration is particularly significant and useful for online computer vision applications like tracking, and image/video retrieval. Once we have learned an image metric, the similarities between other images can be readily measured in accordance with their relations implied by the given training examples.

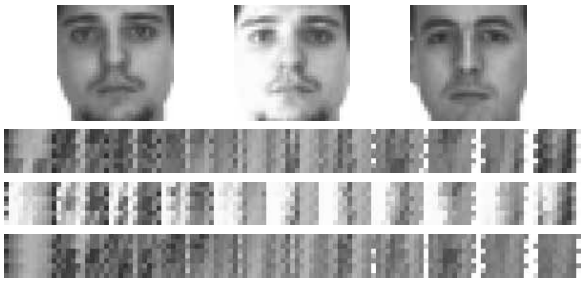

Figure 3: Examples of glocal neighborhood matrices from three face images of two different individuals.

Concerning the side-information in our metric learning, it could either include only the similar image pairs or both the similar and dissimilar ones. Assume that we have a set of paired examples, denoted as $\{(\mathcal{I}_A, \mathcal{I}_B)\} = \mathcal{S} \cup \mathcal{S}'$, where the two sets $\mathcal{S}$ and $\mathcal{S}'$ give the information about these image pairs being similar or dissimilar:

$$\begin{aligned} (\mathcal{I}_A, \mathcal{I}_B) \in \mathcal{S} \quad &\Longleftrightarrow \quad \mathcal{I}_A \text{ and } \mathcal{I}_B \text{ are similar,} \\ &\text{and} \\ (\mathcal{I}_{A'}, \mathcal{I}_{B'}) \in \mathcal{S}' \quad &\Longleftrightarrow \quad \mathcal{I}_{A'} \text{ and } \mathcal{I}_{B'} \text{ are dissimilar.} \end{aligned}$$

Our overall task is to derive an image metric that explains the pairwise relations in $\mathcal{S}$ and $\mathcal{S}'$, and also gives reasonable distance measurements for all image samples. In practice, it would be more flexible to establish a *pseudo-metric* distance function instead of a metric one in that the learning of image metrics often involves dimension reduction of the data. (Note that a pseudo-metric satisfies all properties of a metric except that it allows the distance between two different data to be zero.) Therefore bear in mind that we are to require the learned function to be a pseudo-metric, although hereafter we still refer it as a *metric*.

Let $A, B \in \mathbb{R}^{d \times n}$ be the glocal neighborhood matrices of images $\mathcal{I}_A$ and $\mathcal{I}_B$. Then given two arbitrary rectangular matrices $U \in \mathbb{R}^{d \times \ell}$ ($\ell \leq d$) and $V \in \mathbb{R}^{n \times m}$ ($m \leq n$), we can define the following image metric

$$\begin{aligned} \rho(A, B; U, V) &= \|U^T A V - U^T B V\|_F \\ &= \|U^T (A - B) V\|_F \,. \end{aligned} \tag{1}$$

where $\|\cdot\|_F$ is the Frobenius matrix norm. The proof of the function ρ in (1) being a pseudo-metric is straightforward since every normed linear space is also a metric space.

Clearly the goodness of a metric defined by (1) depends on the bilinear transform consisting of U and V. To gain insight into this issue, we first consider a formulation that the learning of a best metric ρ is based on side-information provided by paired examples only in $\mathcal{S}$. Specifically, we seek a bilinear transform restricted to orthogonal matrices

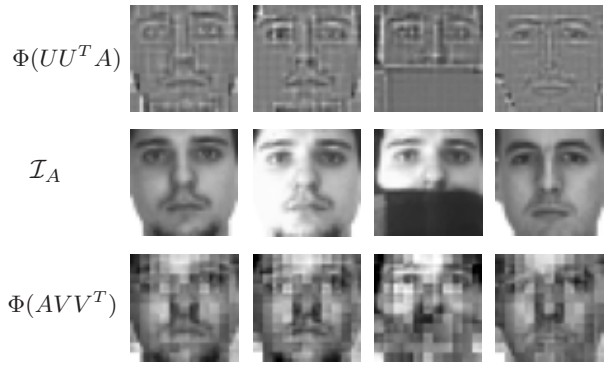

Figure 4: Reconstruction outcomes of *one-sided* bilinear transforms, derived by solving (2) with either $V = I_n$ or $U = I_d$. Here Φ is simply a matrix reshaping to restore a glocal matrix back to its corresponding image matrix.

of U and V, i.e., $U^T U = I_\ell$ and $V^T V = I_m$, that solves

$$\min_{U,V} \sum_{(\mathcal{I}_A, \mathcal{I}_B) \in \mathcal{S}} |\rho(A, B; U, V)|^2 \,. \tag{2}$$

It should be pointed out that incorporating a bilinear transform into the definition of image metric ρ in (1) would not yield the same effectiveness, if the pairwise images were not represented in the glocal form. Recall that a glocal neighborhood matrix induces two structure-meaningful vector spaces: its row space is spanned by vectors pertaining to global image features, and the column space by vectors related to the local ones. To further understand the properties of ρ, we now investigate the roles of U and V.

- That the column space of $U^T A$ is of a lower dimension ($\ell \leq d$) implies only those significant/discriminative features presented in every local area are extracted. That is, U functions like a filter to screen out unnecessary information or variations due to noise or illumination (in the provided side-information). The effect of U is illustrated in the first row of Fig. 4.

- Similarly, since $m \leq n$, a lower dimension of the row space of AV indicates feature extraction is performed to emphasize global image-characteristics shared by the pairwise images in $\mathcal{S}$. Indeed V operates by giving different weights to the block positions, and it is useful for handling occlusions, as shown in Fig. 4.

Alternatively, the side-information used in learning the metric ρ can include pairwise examples both from $\mathcal{S}$ and $\mathcal{S}'$. In this case, we do not require U and V to be orthogonal matrices, and an optimal image metric ρ is obtained by

solving the following constrained optimization problem:

$$\begin{aligned} &\min_{U,\,V} \sum_{(\mathcal{I}_A,\mathcal{I}_B)\in\mathcal{S}} |\rho(A,B;U,V)|^2\,, \\ \text{subject to } &\sum_{(\mathcal{I}_{A'},\mathcal{I}_{B'})\in\mathcal{S}'} |\rho(A',B';U,V)|^2 = c\,, \end{aligned} \tag{3}$$

where c is a constant.

4. Algorithm: Bilinear-Glocal (BiGL) Image Metrics

Since the optimization problems in (2) and (3) employ the Frobenius norm, and both conform to the *general two-sided Procrustes problem* [10], they can be numerically solved by a flip-flop algorithm [14]. To simplify our discussion, we shall give a detailed formulation only on the developing of the algorithm for (2), and succinctly discuss the other.

To begin with, we note that the squared Frobenius norm can be expressed as the matrix trace: $\|U^TAV\|_F^2 = \mathrm{tr}(U^TAVV^TA^TU) = \mathrm{tr}(V^TA^TUU^TAV)$. Now by a flip-flop algorithm for (2), we repeatedly fix one of the U and V while solving the other.

V-Step: Let $\tilde{U}$ be the current estimate of U. Rewrite (2) and solve V in the following optimization problem:

$$\min_{V^TV=I} \sum_{(\mathcal{I}_A,\mathcal{I}_B)\in\mathcal{S}} \mathrm{tr}(V^T(A-B)^T\tilde{U}\tilde{U}^T(A-B)V)\,. \tag{4}$$

U-Step: Similarly, with the estimate $\tilde{V}$, we can solve the following optimization problem for U:

$$\min_{U^TU=I} \sum_{(\mathcal{I}_A,\mathcal{I}_B)\in\mathcal{S}} \mathrm{tr}(U^T(A-B)\tilde{V}\tilde{V}^T(A-B)^TU)\,. \tag{5}$$

Because the matrix trace is a linear function, the summation can be moved inside the trace in (4) and (5). Also, let

$$D_u = \sum_{(\mathcal{I}_A,\mathcal{I}_B)\in\mathcal{S}} (A-B)^T\tilde{U}\tilde{U}^T(A-B)\,, \tag{6}$$

and analogously,

$$D_v = \sum_{(\mathcal{I}_A,\mathcal{I}_B)\in\mathcal{S}} (A-B)\tilde{V}\tilde{V}^T(A-B)^T. \tag{7}$$

Then, by introducing the Lagrange multipliers and by differentiating the objective functions and constraints, we accordingly solve the following two eigenvalue problems to optimize (4) and (5):

$$D_u\mathbf{v} = \lambda\mathbf{v} \quad \text{and} \quad D_v\mathbf{u} = \lambda\mathbf{u}\,. \tag{8}$$

Solving $D_u\mathbf{v} = \lambda\mathbf{v}$ for the eigenvectors $\{\mathbf{v}_i | i = 1,\ldots,m\}$ that correspond to the m smallest eigenvalues, we obtain the solution $\tilde{V} = [\mathbf{v}_1 \cdots \mathbf{v}_m]$ for the optimization problem (4). Likewise, solving $D_v\mathbf{u} = \lambda\mathbf{u}$ for the eigenvectors $\{\mathbf{u}_i | i = 1,\ldots,\ell\}$ that correspond to the ℓ smallest eigenvalues, we have the solution $\tilde{U} = [\mathbf{u}_1 \cdots \mathbf{u}_\ell]$ for the optimization problem (5). To sum up, our algorithm iteratively solves the two eigenvalue problems in (8) to find V and U for the bilinear-glocal (BiGL) image metric.

When the dissimilarity constraints are available as in (3), analogous to (6) and (7), we denote D'_u and D'_v for $(\mathcal{I}_{A'},\mathcal{I}_{B'}) \in \mathcal{S}'$. Following the same line of derivation, the optimization problem (3) is reduced to alternately solving the two generalized eigenvalue problems:

$$D_u\mathbf{v} = D'_u\lambda\mathbf{v} \quad \text{and} \quad D_v\mathbf{u} = D'_v\lambda\mathbf{u}\,. \tag{9}$$

5. Experimental Results

We test our method on the nearest-neighbor classification problems and multi-object tracking. Totally, there are seven datasets used for verifying the effectiveness of the BiGL image metrics in describing the correct similarity relations. As to the multi-object visual tracking, we integrate the BiGL image metric with particle filters to track three targets under dim and varied lighting conditions.

5.1. Nearest-Neighbor Classifications

In each experiment of nearest-neighbor classifications, we apply the BiGL algorithm to learn the underlying image metric with 10 or 20 pairs of images as the side-information. Then, for each test image, we use the learned metric to find its nearest neighbor in the dataset, and check whether their labels are the same. (The class labels can be assigned beforehand according to the classification task.) Hence the performance of a BiGL image metric can be evaluated by the accuracy of the nearest neighbor classifications. Notice that the image pairs in $\mathcal{S}$ and $\mathcal{S}'$ for metric learning are *randomly* selected from the dataset based on the labels. And each reported error rate is indeed the average over ten runs of classifications by the metric learned with different $\mathcal{S}$ and $\mathcal{S}'$. Described below are those datasets we use to build various types of classification problems.

- **The AR Database.** The AR database [12] contains face images that are taken during two sessions. Some typical images of the AR database are shown in Fig. 5. We choose the images of the first session as the AR$_{all}$ dataset. This dataset includes 1,534 ($= 118 \times 13$) images of 118 people. The classification task is to recognize people under illumination changes, variations in facial expression, and occlusions. In addition, we also select the four images of each person with the changes

Figure 5: The AR database contains thousands of face images with different facial expressions, illumination conditions, and occlusions. We use it to generate three datasets, AR_{light}, $\text{AR}_{light+exp}$, and AR_{all}, for our experiments. Each dataset covers certain types of variations.

$\mathcal{S}$ $\mathcal{S}'$

Figure 6: Examples in the similarity and dissimilarity constraints for the $\text{AR}_{light+exp}$ dataset. Each column here represents an instance of intended pairwise relation.

only in illuminations, i.e., other variations are not included. We name the resulting dataset AR_{light}, which consists of 472 images of 118 people. The corresponding task is to recognize people under different lighting conditions. Furthermore, from the AR database, we select 826 ($= 118 \times 7$) face images as the $\text{AR}_{light+exp}$ dataset for the face recognition problem on illumination and expression changes. In Fig. 6, we show some examples of similar and dissimilar pairs that are used for the $\text{AR}_{light+exp}$ task.

- **YaleB Face Database.** The YaleB face database [5] is a widely-used benchmark for face recognition. It contains 5,760 single light source images of 10 people. We choose to use only the frontal-view images, and construct a dataset of 640 images. The task is also to identify the same person under lighting changes.

- **Caltech Face Images.** We select 125 face images of 25 people from the Caltech-101 object categories [3]. Different from AR and YaleB, the dataset contains face images taken under uncontrolled or natural lighting conditions. The classification task is again to identify the subjects. In Fig. 7, we show the five images of the first subject in the dataset.

- **AR_{light} Gender.** The preceding AR_{light} includes 260 face images of 65 males and the other 212 of 53 females. Instead of recognizing people, this experiment is to test the effectiveness of the BiGL metric on differentiating between the female and the male faces.

Figure 7: We choose a subset of face images from the Caltech-101 object categories, and use it for the experiment of recognizing faces with uncontrolled lighting conditions. There are five images from each of the 25 people.

- **Natural Textures.** We download several webcam video clips [13] from the Web to generate a dataset for the experiment on classifying various textures. Textures of the same type are cropped from the same position in the video frames, taken under varying weather and illumination conditions. Fig. 8 shows two different classes of textures in the dataset, which contains 120 texture images of 20 classes.

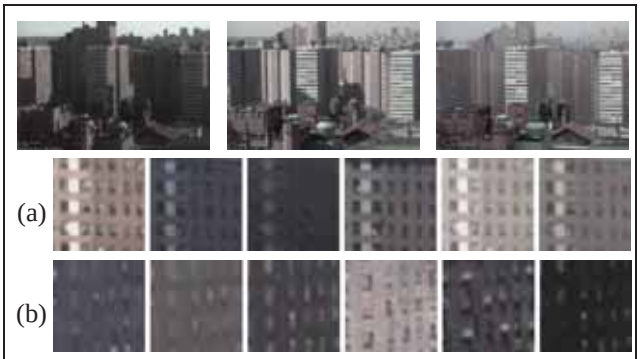

Figure 8: Classifying 20 types of texture patterns. The images in (a) and (b) belong to two different types of texture patterns. Textures of the same type are cropped from the same position in the video frames, which are captured under different weather and illumination conditions (as shown in the top row).

For all the aforementioned experiments, we compare the BiGL metrics with the Mahalanobis distances computed by the *Relevant Component Analysis* (RCA) [16], as well as the Euclidean distance. (RCA computes the whitening transformation according to the equivalence relations of data.) The classification outcomes are listed in Table 1. Throughout this work we use only the gray-level information, and resize all images in the experiments to 33×33 pixels. Therefore, for RCA and the Euclidean distance, the input vectors have 1089 dimensions; for BiGL, the glocal matrices are of size 9×121, where we choose $h = 3$. As listed in the table, the dimensionality is reduced to 2×40 (80 for RCA) and 3×50 (150 for RCA), i.e., we have $\ell = 2, 3$ and $m = 40, 50$. (The setting is to make the comparisons fair.) For convenience,

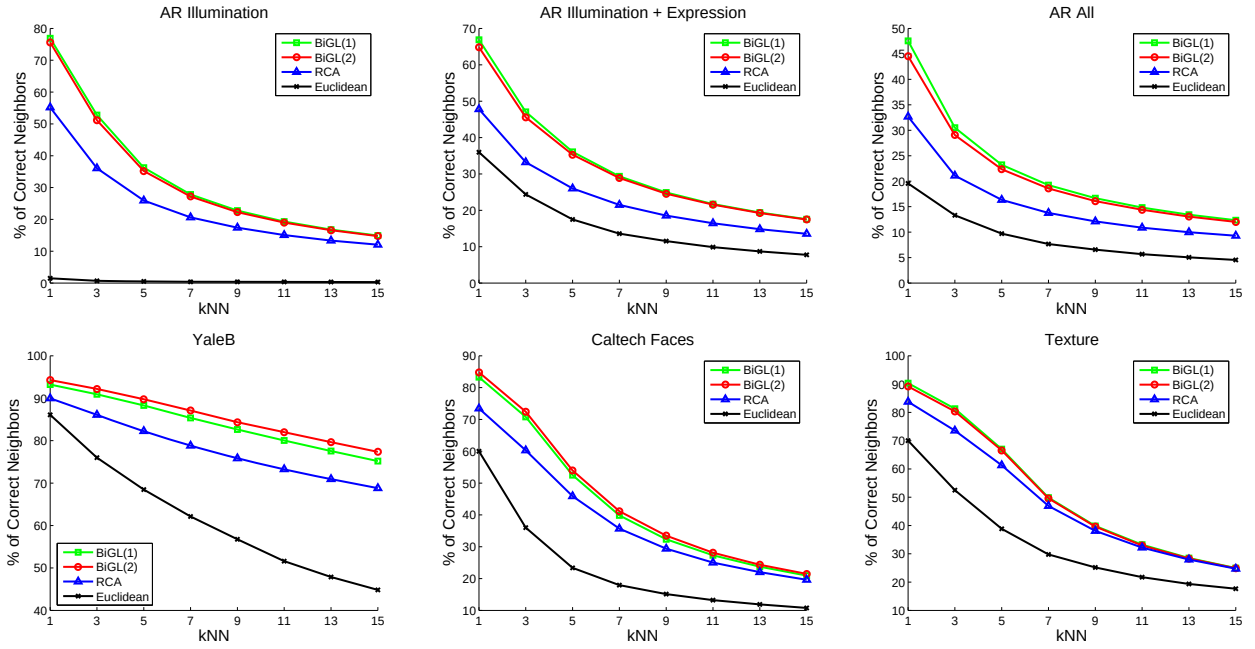

Figure 9: We compute the percentages of the correct neighbors in the k-nearest neighborhood, with $k = 1, 3, \ldots, 15$. The results are ten-run averages for AR_{light}, $\text{AR}_{light+exp}$, AR_{all}, YaleB, Caltech faces, and the texture dataset. The numbers of pairwise examples are $|\mathcal{S}| = |\mathcal{S}'| = 10$, and the number of reduced dimensions is 80 ($\ell = 2$, $m = 40$).

we write BiGL(1) to denote the BiGL metric learned with $\mathcal{S}$, and BiGL(2) with both $\mathcal{S}$ and $\mathcal{S}'$ (see equations (2) and (3)). Our results indicate that BiGL and RCA both improve the Euclidean distance for nearest neighbor classifications. In particular, the proposed BiGL sufficiently reduces the dimensionality while improving the accuracy of classifications. When the number of examples is small, the BiGL metrics are more effective than RCA in correlating the distances between image data with their similarity relations.

For a more detailed analysis, we compute the percentages of the correct neighbors in the k-nearest neighborhood with $k = 1, 3, \ldots, 15$ for all datasets. Six of the results are illustrated in Fig. 9, where $|\mathcal{S}| = |\mathcal{S}'| = 10$ and the number of reduced dimensions is 80 ($\ell = 2$, $m = 40$). The results are consistent with the performances of RCA and BiGL in the nearest-neighbor classifications.

Besides the comparisons summarized in Table 1, we have also tested the *Relief* and the *Simba* feature selection algorithms [6] for metric learning. The testings are carried out by directly using the Web-available MATLAB code [6] for all the foregoing experiments. However, as these two methods are aimed at selecting discriminative dimensions in the original space rather than mapping the data to another subspace (like RCA), *Relief* and *Simba* improve the Euclidean distance marginally, and do not achieve comparable error-rates to those of RCA and BiGL.

Indeed, when we are allowed to more carefully select the dissimilar examples of the side-information, the error rates by BiGL(2) might be further decreased. For some specific problems, e.g., AR_{light}, we can easily single out the unwanted factor (i.e., the lighting change) for comparing two images. Instead of choosing at random, we build the dissimilarity constraints by picking out the pairs of faces that are of different people but with the same lighting condition. The column of AR_{light} in Table 1 can then be improved as $24.39\%\rightarrow 22.76\%$, $20.24\%\rightarrow 18.19\%$, $15.15\%\rightarrow 14.44\%$, and $14.13\%\rightarrow 13.73\%$.

We note that if only a few examples are available for learning, the rank of the covariance matrix for RCA should be very low, compared with the dimensionality of the input space. Sine RCA computes the whitening transformation based on the inverse of the covariance matrix, the whitening weights derived from a low-rank covariance are probably inaccurate due to the existence of many nearly-zero singular values. Although using PCA as a preprocessing step might alleviate the situation (as presented in [16] for dimensionality reduction), to compute the total scatter matrix of PCA from very few examples may still be inaccurate. On the other hand, BiGL suffers less from the singularity issue for the following two reasons: 1) The glocal image representation ensures that the number of the dimensions associated with the bilinear transform is small (e.g., $d = 9$

and $n = 121$). Furthermore, the mechanism of dimensionality reduction is inherent in the BiGL formulation, no data-preprocessing step is needed. 2) Even with the singularity, the optimization problem (2) can still be solved. (In addition to the effective dimensions, we just need to choose arbitrary orthogonal bases satisfying $U^T U = I_\ell$ and $V^T V = I_m$ for the singular dimensions.) When applying the metric to the test data, the distances measured in the singular dimensions will keep unchanged to let the effective dimensions dominate the measurement, whereas RCA will give inaccurate whitening weights to the singular dimensions and thus deteriorate the measurement of the effective dimensions.

5.2. Multi-Object Tracking

The efficiency of learning an image metric from a few examples allows us to apply the BiGL algorithm to online applications like visual tracking. It has been shown in the seminal work of [17] that metrics can be integrated in an exemplar-based probabilistic paradigm for tracking. In our experiment, we use a simpler setting to test the BiGL metrics. The BiGL learning algorithm is combined with the particle filters to simultaneously track multiple targets in a video sequence. During the tracking with particle filters, we collect the image patches that are of high observation probability, and generate the pairwise examples for $\mathcal{S}$ and $\mathcal{S}'$. The set $\mathcal{S}$ therefore contains image pairs of the same target, and $\mathcal{S}'$ could include pairs of image samples of different targets, or of a target and nearby background. In general, we maintain ten pairs for each set ($|\mathcal{S}| = |\mathcal{S}'| = 10$), and periodically replace some previous pairs in $\mathcal{S}$ and $\mathcal{S}'$ with new ones. The first few frames are tracked using the Euclidean distance. Once $\mathcal{S}$ and $\mathcal{S}'$ are ready, we construct the BiGL metric for all targets, and thereafter use it in the observation likelihood of the particle filters. More specifically, the observation likelihood includes the BiGL metric by

$$p(z|X) \propto \frac{1}{Z} \exp -\gamma\, \rho\big(A(\mathcal{I}_z), B(X, \mathcal{I}_0); U, V\big)\,, \quad (10)$$

where $A(\mathcal{I}_z)$ is the glocal matrix of the observed image patch $\mathcal{I}_z$, and $B(X, \mathcal{I}_0)$ is the glocal matrix of the target template $\mathcal{I}_0$ with the transformation hypothesized by the particle state X. We test this tracking algorithm on a video sequence with a dim lighting condition. Two sample frames are displayed in Fig. 10, and the tracking results of the respective frames are shown in the bottom row. Note that the brightness and the contrast of these face images are enhanced just for better displaying. In our experiment, we do not apply any image enhancement operator to the image frames. With the learned metrics, our algorithm successfully tracks the three faces throughout the whole of 195 frames, using 500 particles for each target. (See the supplementary video for the complete tracking result.)

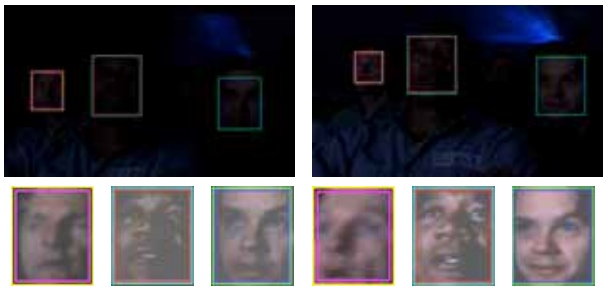

Figure 10: The top row includes two sample frames from the video sequence for our experiment on multi-object tracking. The targets in the respective frames are shown in the bottom row. Note that, in this figure, we enhance the brightness and the contrast of the face images for better displaying; however, we do not apply any image enhancement operator to the image frames in our experiment.

6. Conclusions

We have described a new method for learning an image metric based on a few pairwise examples. Throughout our work there are two main concepts: *image representation* and *bilinear-transform optimization*. The motivation for proposing a new image representation is prompted by the inefficiencies and unsatisfactory results of testing by directly applying the bilinear transform to the canonical form of an image matrix. On the contrary, to respectively embed the local and global image properties into the column and row spaces of a glocal matrix is very convenient for the analysis of bilinear transform, and indeed greatly enhances the effectiveness of a resulting image metric.

Overall, the proposed BiGL metric learning algorithm finds an optimal bilinear transform for image data to best explain the pairwise similarity or dissimilarity relations, given as the side-information. With the glocal image representation, the optimal bilinear transform takes advantage of the intrinsic global and local image features to yield an effective image metric. We have also pointed out when the number of data is small, solving an optimization problem of a high dimensionality is prone to numerical instability due to the rank deficiency and the matrix singularity. Since the BiGL algorithm only needs to solve an optimization problem of a lower dimensionality (benefitted from the glocal representation), the offline or online learning process is thus computationally more stable and more efficient.

Acknowledgements

This work was supported by grants 93-2213-E-001-010 and 94-EC-17-A-02-S1-032.

Table 1: The Average Error Rates (%) of the Nearest Neighbor Classifications.

Method	Dimensions	Number of pairs	AR_{light}	$\text{AR}_{light+exp}$	AR_{all}	YaleB	Caltech	AR_{light} Gender	Texture
Euclidean	1089	—	98.51	64.04	80.44	13.90	40.00	10.80	30.00
RCA	80	10	44.80	52.20	67.321	10.03	26.56	8.87	16.25
BiGL(1)	2×40	10	*23.08*	*33.13*	*52.44*	6.78	16.72	5.57	*9.66*
BiGL(2)	2×40	$10 + 10$	24.39	35.16	55.44	*5.71*	*15.20*	*5.38*	10.83
RCA	150	10	38.03	44.81	61.16	7.81	20.72	7.45	13.83
BiGL(1)	3×50	10	*18.91*	*27.16*	*44.83*	*3.17*	13.60	4.70	*8.00*
BiGL(2)	3×50	$10 + 10$	20.24	29.17	48.26	4.45	*13.52*	*4.21*	8.75
RCA	80	20	23.24	35.41	57.26	5.75	13.20	5.80	9.41
BiGL(1)	2×40	20	17.34	24.83	44.68	4.54	*8.32*	5.93	8.75
BiGL(2)	2×40	$20 + 20$	*15.15*	*24.07*	*41.87*	*3.70*	*8.32*	*4.66*	*7.08*
RCA	150	20	20.65	31.05	51.57	5.01	11.20	5.21	8.33
BiGL(1)	3×50	20	14.72	22.35	37.48	*2.81*	8.08	4.55	7.41
BiGL(2)	3×50	$20 + 20$	*14.13*	*21.89*	*34.89*	*2.81*	*6.64*	*3.91*	*6.66*

The number of pairs indicates $|\mathcal{S}|$ for RCA and BiGL(1), and $|\mathcal{S}| + |\mathcal{S}'|$ for BiGL(2). Each reported error rate is the average over ten runs of classifications by the metrics learned with randomly-selected $\mathcal{S}$ and $\mathcal{S}'$.

References

[1] V. Athitsos, J. Alon, S. Sclaroff, and G. Kollios, "BoostMap: a Method for Efficient Approximate Similarity Rankings," *CVPR*, vol. 2, pp. 268–275, 2004.

[2] L. Fei-Fei, R. Fergus, and P. Perona, "A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories," *ICCV*, vol. 2, pp. 1134–1141, 2003.

[3] L. Fei-Fei, R. Fergus, and P. Perona, "Learning Generative Visual Models from Few Training Examples: an Incremental Bayesian Approach Tested on 101 Object Categories," *CVPR Workshop on Generative-Model Based Vision*, 2004.

[4] M. Fink, "Object Classification from a Single Example Utilizing Class Relevance Pseudo-Metrics," *NIPS 17*, pp. 449–456, 2004.

[5] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman, "From Few To Many: Generative Models For Recognition Under Variable Pose and Illumination," *AFGR*, pp. 277–284, 2000.

[6] R. Gilad-Bachrach, A. Navot, and N. Tishby, "Margin Based Feature Selection – Theory and Algorithms," *ICML*, 2004.

[7] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood Components Analysis," *NIPS 17*, pp. 513–520, 2004.

[8] T. Hertz, A. Bar-Hillel, and D. Weinshall, "Learning Distance Functions for Image Retrieval," *CVPR*, vol. 2, pp. 570–577, 2004.

[9] T. Hertz, N. Shental, A. Bar-Hillel, and D. Weinshall, "Enhancing Image and Video Retrieval: Learning via Equivalence Constraint," *CVPR*, vol. 2, pp. 668–674, 2003.

[10] N. Higham, "Matrix Procrustes Problems," Tech. Rep., Department of Mathematics, University of Manchester, 1994.

[11] S. Mahamud and M. Hebert, "Minimum Risk Distance Measure for Object Recognition," *ICCV*, pp. 242–248, 2003.

[12] A.M. Martinez and R. Benavente, "The AR Face Database," Tech. Rep. CVC Technical Report #24, Computer Vision Center at the U.A.B, June 1998.

[13] S.G. Narasimhan, C. Wang, and S.K. Nayar, "All the Images of an Outdoor Scene," *ECCV*, vol. 3, pp. 148–162, 2002.

[14] P.H. Schonemann, "On Two-Sided Orthogonal Procrustes Problems," *Psychometrika*, vol. 33, pp. 19–33, 1968.

[15] A. Shashua and L. Wolf, "Kernel Feature Selection with Side Data Using a Spectral Approach," *ECCV*, vol. 3, pp. 39–53, 2004.

[16] N. Shental, T. Hertz, D. Weinshall, and M. Pavel, "Adjustment Learning and Relevant Component Analysis," *ECCV*, vol. 4, pp. 776–792, 2002.

[17] K. Toyama and A. Blake, "Probabilistic Tracking in a Metric Space," *ICCV*, vol. 2, pp. 50–57, 2001.

[18] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl, "Constrained K-Means Clustering with Background Knowledge," *ICML*, pp. 577–584, 2001.

[19] L. Wolf and A. Shashua, "Feature Selection for Unsupervised and Supervised Inference: the Emergence of Sparsity in a Weighted-Based Approach," *ICCV*, pp. 378–384, 2003.

[20] E.P. Xing, A.Y. Ng, M.I. Jordan, and S.J. Russell, "Distance Metric Learning with Application to Clustering with Side-Information," *NIPS 15*, pp. 505–512, 2002.