Learning IMED via Shift-Invariant Transformation

SUN Bing, FENG Jufu and WANG Liwei
Key Laboratory of Machine Perception (Peking University), MOE
Department of Machine Intelligence
School of Electronics Engineering and Computer Science
Peking University, Beijing 100871, P.R.China
{sunbing,fjf,wanglw}@cis.pku.edu.cn

Abstract

The IMage Euclidean Distance (IMED) is a class of image metrics, in which the spatial relationship between pixels is taken into consideration. It was shown that calculating the IMED of two images is equivalent to performing a linear transformation called Standardizing Transform (ST) and then followed by the traditional Euclidean distance. However, while the IMED is invariant to image shift, the ST is not a Shift-Invariant (SI) filter. This left as an open problem whether IMED is equivalent to SI transformation plus traditional Euclidean distance. In this paper, we give a positive answer to this open problem. Specifically, for a wider class of metrics, including IMED, we construct closed-form SI transforms. Based on the SI metric-transform connection, we next develop an image metric learning algorithm by learning a metric filter *in the transform domain. This is different from all previous metric approaches. Experimental results on benchmark datasets demonstrate that the learned image metric has promising performances.*

1. Introduction

Determining a distance measure over the images is a fundamental problem in computer vision and pattern recognition. The distance metric can be either *learned* from training data, or *designed* according to prior knowledge. The former, namely the problem of *metric learning*, has gained great interest in recent years [14, 26, 1, 5, 3, 4, 24, 7, 12, 23, 25]; and the latter, which we refer to as the problem of *metric design*, is often more difficult [10, 8, 19, 13, 21] because the "prior" knowledge are usually hard to obtain and express.

The *IMage Euclidean Distances (IMED)* [21] is a class of image metrics, which tries to deal with the problem that the relative pixel locations are not considered in the traditional *Euclidean distance*. The traditional Euclidean distance sometimes yields counter intuitive result that a perceptually large distortion produces a distance smaller than that produced by a small deformation. IMED, by merging the special information of the pixels, largely solves this problem. The key advantage of IMED is that it can be embedded in most classification techniques such as SVM, LDA, and PCA. Experiments and applications demonstrated significant performance improvement in many real world problems [21, 2, 17, 18, 20, 22, 6, 27].

Embedding IMED into SVM, LDA *etc*. can be done by involving a linear transformation on the images. It was shown that calculating the IMED of two images is equivalent to performing a linear transformation called Standardizing Transform (ST) and then followed by the traditional Euclidean distance. Hence, feeding the (ST-)transformed images to a recognition algorithm automatically embeds IMED in it. From the transformation point of view, the authors of [21] argued that IMED actually smoothes the images. This seems established a connection between image metrics and transformation domain processing.

Another property of IMED is its invariance to image shift. That is, if one performs the same image shift (translation *etc*.) to two images, their IMED remains invariant[1]. However, while IMED is a shift invariant metric, the associated Standardizing Transform (ST) does not have this property, i.e. ST is not a shift invariant (SI) transformation. This left an open problem whether IMED is equivalent to a SI filter plus traditional Euclidean distance.

In this paper, we first give a positive answer to this open problem. We construct, for every IMED, a closed-form SI transform. In fact, we establish a relationship between the shift-invariant metric and the SI transform.

We next consider the metric learning problem. Based on the metric-transformation connection, we learn an image metric by learning a *metric filter* in the transform domain. This is different from all previous metric learning

[1] The formal definition of *shift-invariance* will leave in the next section.

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

approaches. Experimental results on benchmark datasets demonstrate that the learned metric has promising performances.

The rest of this paper is organized as follows: Section 2 presents a brief review of IMED and related work. In Section 3, for every SI metric, we give a closed-form construction of SI transform. Based on the results of Section 3, we develop a metric learning algorithm in Section 4. Section 5 presents experimental results. Finally, a conclusion is given in Section 6.

2. A Brief Review of IMED and related work

It is convenient to denote an image X of size $n_1 \times n_2$ as a vector $\boldsymbol{x} = \mathrm{vec}(X)$, and the $(n_2 i_1 + i_2)$-th component of $\boldsymbol{x}$ is the intensity at the (i_1, i_2) pixel. Here $\mathrm{vec}(X)$ is the *vectorization* of X; specifically, $\mathrm{vec}(X)$ is the $n_1 n_2 \times 1$ column vector obtained by stacking the rows of X.

The standard Euclidean distance $d_E(\boldsymbol{x}, \boldsymbol{y})$ is

$$d_E(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\sum_{k=1}^{n_1 n_2} (x_k - y_k)^2} = \sqrt{(\boldsymbol{x} - \boldsymbol{y})^T (\boldsymbol{x} - \boldsymbol{y})},$$

which does not take into consideration the spatial relationships of pixels, probably leading to undesired results [10, 21]. To solve the problem, Wang *et al.*[21] proposed the IMED d_G, defined as:

$$\begin{aligned} d_G(\boldsymbol{x}, \boldsymbol{y}) &= \sqrt{\sum_{i,j=1}^{n_1 n_2} g_{ij} (x_i - y_i)(x_j - y_j)} \\ &= \sqrt{(\boldsymbol{x} - \boldsymbol{y})^T G (\boldsymbol{x} - \boldsymbol{y})}. \end{aligned}$$

The $n_1 n_2 \times n_1 n_2$ metric matrix G solely determines the IMED, where the element g_{ij} represents how the "seat"[2] x_i affects the "seat" x_j. Replacing G with the identity matrix, we get the standard Euclidean distance. The main constraints for IMED are that the element g_{ij} depends only on the pixel distance between pixels P_i and P_j, that is $g_{ij} = f(\|P_i - P_j\|)$, and that g_{ij} monotonically decreases as $\|P_i - P_j\|$ increases. The reader should be aware that the pixel distance $\|P_i - P_j\|$, which is the distance between P_i and P_j on the image lattice, is different from the image distance d_G measured in the high dimensional image space. A constraint on f is that it must be a continuous positive definite function, thereby ensuring that G is positive definite.

Any f with the above properties defines an IMED. A typical example is that $f(\cdot)$ is the Gaussian function ([21]), i.e. the metric coefficients are

$$\begin{aligned} g_{ij} &= f(\|P_i - P_j\|) \\ &= \frac{1}{2\pi\sigma^2} e^{-\frac{|P_i - P_j|^2}{2\sigma^2}} \\ &= \frac{1}{2\pi\sigma^2} e^{-\frac{(i_1 - j_1)^2 + (i_2 - j_2)^2}{2\sigma^2}}, \end{aligned} \quad (1)$$

where $P_i = (i_1, i_2), P_j = (j_1, j_2)$.

As suggested in [21], the calculation of IMED can be simplified by decomposing G to $A^T A$, leading to

$$\begin{aligned} d_G^2(\boldsymbol{x}, \boldsymbol{y}) &= (\boldsymbol{x} - \boldsymbol{y})^T G (\boldsymbol{x} - \boldsymbol{y}) \\ &= (\boldsymbol{x} - \boldsymbol{y})^T A^T A (\boldsymbol{x} - \boldsymbol{y}) \\ &= (\boldsymbol{u} - \boldsymbol{v})^T (\boldsymbol{u} - \boldsymbol{v}), \end{aligned}$$

where $\boldsymbol{u} = A\boldsymbol{x}, \boldsymbol{v} = A\boldsymbol{y}$. The Standardizing Transform (ST) is the special case when $A^T = A$, commonly written as $A = G^{\frac{1}{2}}$. By the eigen-decomposition of $G^{\frac{1}{2}}$, Wang *et al.* showed that ST is actually a transform domain smoothing [21].

A very important property of IMED is that it can be easily embedded into most image recognition algorithms. That is, feeding the ST-transformed image $\boldsymbol{u} = G^{\frac{1}{2}} \boldsymbol{x}$ to a recognition algorithm automatically embeds IMED in it.

Another image metric, called the Generalized Euclidean Distance (GED), was proposed for binary images in [10]. IMED and GED are quite similar except the distance measure between pixels P_i and P_j. Specifically, the metric matrix for GED is defined as a Laplacian function:

$$g_{ij} = e^{-\alpha \cdot (|i_1 - j_1| + |i_2 - j_2|)}, \quad (2)$$

where α is a scale parameter.

As pointed out in [21], shift-invariance (SI) is a necessary property for any intuitively reasonable image metric. We use the terminology *SI metric* as in [21], which is consistent with the *shift-invariant operator* [15] in signal processing. Mathematically, for any images X, Y, a distance measure $d(\cdot, \cdot)$ is shift invariant if and only if

$$d(X, Y) = d(X_\tau, Y_\tau),$$

where X_τ, Y_τ is an image shift of X, Y, respectively.

Another common meaning of *shift-invariance* requires that

$$d(X, Y) = d(X, Y_\tau),$$

as in the case of the tangent distance [19]. Such distance measures usually cannot meet the requirements of the metric axioms, so they are beyond the scope of this paper.

Note that IMED (and GED) depends only on the relative position between pixels P_i and P_j, i.e. $g_{ij} = g[i_1 - j_1, i_2 - j_2]$, where $i = n_2 i_1 + i_2, j = n_2 i_1 + i_2$. This makes g_{ij} invariant to image shift.

[2] The word "seat" is used to emphasize that g_{ij} depends only on the position of x_i and x_j, but is independent on the value of x_i and x_j.

However, while IMED is an SI image metric, the associated Standardizing Transform is not an SI transformation. In fact, SI transformations are not only important in real applications but also play a central role in image processing. Since ST is not an SI transform, this left an open problem whether IMED can be decomposed to SI transformations. That is, for an IMED metric matrix G, does there exist an SI transform H such that

$$G = H^T H.$$

3. The SI Transform of an SI Metric

In this section, we give a positive answer to the open problem. Actually we solve a more general problem: we show that any SI image metric can be decomposed to SI transform plus traditional Euclidean distance.

3.1. Notations

Since the $n_1 n_2 \times n_1 n_2$ metric matrix G is an SI metric, there exists a function $g[i_1, i_2]$, such that

$$G(i,j) = g[i_1 - j_1, i_2 - j_2],$$

where $i = i_1 n_2 + i_2, j = j_1 n_2 + j_2$.

The metric G can be either separable or inseparable. G is said to be separable if there exist $g_1[i], g_2[i]$ such that

$$g[i_1 - j_1, i_2 - j_2] = g_1[i_1 - j_1] \cdot g_2[i_2 - j_2].$$

A metric is said to be inseparable if is not separable. Because

$$e^{-\frac{(i_1-j_1)^2+(i_2-j_2)^2}{2\sigma^2}} = e^{-\frac{(i_1-j_1)^2}{2\sigma^2}} \cdot e^{-\frac{(i_2-j_2)^2}{2\sigma^2}},$$

and

$$e^{-\alpha\cdot(|i_1-j_1|+|i_2-j_2|)} = e^{-\alpha\cdot|i_1-j_1|} \cdot e^{-\alpha\cdot|i_2-j_2|},$$

taking $g_1[i] = g_2[i] = \frac{1}{\sqrt{2\pi}} e^{-\frac{i^2}{2\sigma^2}}$ and $g_1[i] = g_2[i] = e^{-\alpha\cdot|i|}$, both IMED and GED are separable SI metrics.

It is known that a separable G is the Kronecker product (or the tensor product) of G_1 and G_2 [9]:

$$G = G_1 \otimes G_2,$$

where $G_1(i,j) = g_1[i-j], G_2(i,j) = g_2[i-j]$.

Below, we start with the case of the separable SI metric (*e.g.* the case of IMED and GED) to derive an SI transform. The inseparable case involves more complicated tensor notions and will be given in Section 3.3.

3.2. The Separable Case

In this subsection we study the case that the metric is separable. That is

$$g[i_1 - j_1, i_2 - j_2] = g_1[i_1 - j_1] \cdot g_2[i_2 - j_2],$$

and

$$G = G_1 \otimes G_2,$$

where $G(i,j) = g[i_1 - j_1, i_2 - j_2], i = i_1 n_2 + i_2, j = j_1 n_2 + j_2$, and $G_k(i,j) = g_k[i-j]$ $(k = 1, 2)$. For any SI metric matrix G satisfying certain condition, we will show there is an SI linear transformation H, so that $G = H^T H$.

Let $\hat{g}_k(\omega)$ denote the discrete time Fourier transform (DTFT) of $g_k[i]$:

$$\hat{g}_k(\omega) = \sum_{i\in\mathbb{Z}} g_k[i]\, e^{\sqrt{-1}i\omega}.$$

Assuming $\hat{g}_k(\omega) \geqslant 0$ for all ω, define $h_k[i]$ as:

$$h_k[i] = \mathcal{F}^{-1}\left(\sqrt{\hat{g}_k(\omega)}\right), \tag{3}$$

Supposing $g_k[i]$ is supported on $[-m_k, m_k]$, a lemma by Riesz [16] asserts that the support of $h_k[i]$ is a subinterval of $[-m_k, m_k]$, i.e. $h_k[i]$ is at most supported on $[-m_k, m_k]$. Let H_k be the $(n_k + 2m_k) \times n_k$ matrix defined as:

$$H_k(i,j) = \begin{cases} h_k[i-j-m], & \text{if } |i-j-m| \leqslant m, \\ 0, & \text{else,} \end{cases} \tag{4}$$

We present the following theorem.

Theorem 1 *Using the notions defined above, the SI metric G can be decomposed as*

$$G = H^T H,$$

where H is a SI linear transformation given by

$$H = H_1 \otimes H_2,$$

and H_k is given by (4).

Proof We first prove that $G_k = H_k^T H_k$. (3) implies that $\hat{g}_k(\omega) = |\hat{h}_k(\omega)|^2$, thus by Parseval's formula and the convolution theorem [15], we have

$$\begin{aligned} G_k(i,j) &= g_k[i-j] \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{g}_k(\omega)\cdot e^{\sqrt{-1}\omega(i-j)}\mathrm{d}\omega \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |\hat{h}_k(\omega)|^2 \cdot e^{\sqrt{-1}\omega(i-j)}\mathrm{d}\omega \\ &= \frac{1}{2\pi}\left\langle \hat{h}_k \cdot e^{-\sqrt{-1}\omega j}, \hat{h}_k \cdot e^{-\sqrt{-1}\omega i}\right\rangle \\ &= \frac{1}{2\pi}\left\langle \widehat{h_k * \delta_j}, \widehat{h_k * \delta_i}\right\rangle \\ &= \langle h_k * \delta_j, h_k * \delta_i\rangle, \end{aligned}$$

where $\delta_m[i] = \delta_{m,i}$ is the Kronecker delta and $*$ denotes the convolution operation.

The definition (4) of H_k is equivalent to $H_k \boldsymbol{e}_i = h_k * \delta_i$, where $\boldsymbol{e}_i$ is the i-th standard basis vector of $\mathbb{R}^{n_k+2m_k}$. Thus $G_k(i,j) = \boldsymbol{e}_j^T \left(H_k^T H_k\right) \boldsymbol{e}_i$, or $G_k = H_k^T H_k$.

By the properties of Kronecker product [11], the following

$$\begin{aligned} G_1 \otimes G_2 &= \left(H_1^T H_1\right) \otimes \left(H_2^T H_2\right) \\ &= \left(H_1^T \otimes H_2^T\right)(H_1 \otimes H_2) \\ &= (H_1 \otimes H_2)^T (H_1 \otimes H_2) \\ &= H^T H \end{aligned}$$

completes the proof. □

By the above theorem, the squared norm of X with respect to an SI metric G is:

$$\operatorname{vec}(X)^T G \operatorname{vec}(X) = \|H \operatorname{vec}(X)\|^2, \tag{5}$$

where $\|\cdot\|^2$ denotes the traditional Euclidean metric. Define $h[i_1, i_2] = h_1[i_1] h_2[i_2]$, it is known that

$$H_1 X H_2^T = h * X.$$

Since $(H_1 \otimes H_2) \operatorname{vec}(X) = \operatorname{vec}\left(H_1 X H_2^T\right)$, (5) can be computed by

$$\begin{aligned} \operatorname{vec}(X)^T G \operatorname{vec}(X) &= \|H \operatorname{vec}(X)\|^2 \\ &= \|(H_1 \otimes H_2) \operatorname{vec}(X)\|^2 \\ &= \|\operatorname{vec}\left(H_1 X H_2^T\right)\|^2 \\ &= \|h * X\|^2. \end{aligned}$$

That is, an SI metric is equivalent to the convolution by the filter h plus traditional Euclidean distance.

The filter h can be directly computed by

$$h[i_1, i_2] = \mathcal{F}^{-1}\left(\sqrt{\hat{g}(\omega_1, \omega_2)}\right), \tag{6}$$

because

$$\begin{aligned} \hat{h}(\omega_1, \omega_2) &= \mathcal{F}(h_1[i_1] h_2[i_2]) \\ &= \hat{h}_1(\omega_1) \hat{h}_2(\omega_2) \\ &= \sqrt{\hat{g}_1(\omega_1)} \cdot \sqrt{\hat{g}_2(\omega_2)} \\ &= \sqrt{\hat{g}(\omega_1, \omega_2)}. \end{aligned}$$

By Theorem 1 and Eq. (6), the function g uniquely determine the SI metric matrix G and the SI transform H (or h). g plays a role to connect the metric and the transform, and is referred as the *metric filter*.

Compared to the ST decomposition $G^{\frac{1}{2}}$, H is no longer a square matrix; actually, H is of size

$$(n_1 + 2m_1)(n_2 + 2m_2) \times n_1 n_2.$$

This is commonly not a problem in practice. In fact, if $n_k \gg m_k$ or h is decreasing rapidly, truncating H to an $n_1 n_2 \times n_1 n_2$ matrix affects little; even not the case, we can use circular convolution and discrete Fourier transform (DFT) to derive an square transform matrix $\tilde{H}$ that exactly keep the metric G.

The condition $\hat{g}(\omega) \geqslant 0$ is sufficient and necessary for the existence of the SI transform of G. It can be shown that IMED satisfies the above condition. Because the exponentials in IMED and GED are rapidly decaying and thus can be viewed as having a finite support, we have the SI decomposition for them.

3.3. The Inseparable Case

In this subsection, we study the case that the metric is inseparable. The results are also generalized for multi-dimensional inputs, *e.g.* 3-D object represented by an $n_1 \times n_2 \times n_3$ "matrix" [20].

It is convenient to use tensor notations to provide a clear expression of the materials. Tensors can be regarded as a multi-index generalization of the vector concept. The number of indices in the representing array is called the *rank* of a tensor. Thus, scalars are rank zero tensors, vectors are rank one tensors, matrices are rank two tensors and a $n_1 \times n_2 \times n_3$ "matrix" is a tensor of rank three. A tensor of type (p, q) has a rank of $p + q$, written as t_q^p.

The Einstein notation is also used here [11]. According to this convention, when an index variable appears twice in a single term, once in an upper (superscript) and once in a lower (subscript) position, it implies that we are summing over all of its possible values, *e.g.* $c_i x^i$ means $\sum c_i x_i$.

Defining the $(d, 0)$ *input tensor* $\mathbb{x}$ by

$$\mathbb{x}^{\boldsymbol{i}} = x[\boldsymbol{i}],$$

the (d, d) SI *metric tensor* $\mathbb{g}$ (which is *positive* and denoted as $\mathbb{g} \succeq 0$) by

$$\mathbb{g}_{\boldsymbol{j}}^{\boldsymbol{i}} = g[\boldsymbol{i} - \boldsymbol{j}],$$

where $\boldsymbol{i} = (i_1, i_2, \cdots, i_d)$, $\boldsymbol{j} = (j_1, j_2, \cdots, j_d)$ are conventions for clarity, the norm of $\mathbb{x}$ with respect to $\mathbb{g}$ is computed as

$$\|\mathbb{x}\|_{\mathbb{g}} = \sqrt{\mathbb{x}^{\boldsymbol{j}} \mathbb{g}_{\boldsymbol{j}}^{\boldsymbol{i}} \mathbb{x}_{\boldsymbol{i}}}.$$

We need to find an SI transform to implement the metric $\mathbb{g}$ and the following theorem gives a construction of the transform.

Let $\hat{g}[\boldsymbol{\omega}]$ denote the multivariate DTFT of $g[\boldsymbol{i}]$, where $\boldsymbol{\omega} = (\omega_1, \omega_2, \cdots, \omega_d)$. Assuming $\hat{g}[\boldsymbol{\omega}] \geqslant 0$, define $h[\boldsymbol{i}]$ as:

$$h[\boldsymbol{i}] = \mathcal{F}^{-1}\left(\sqrt{\hat{g}(\boldsymbol{\omega})}\right).$$

Let the SI transform tensor $\mathbb{h}_{\boldsymbol{j}}^{\boldsymbol{i}}$ be

$$\mathbb{h}_{\boldsymbol{j}}^{\boldsymbol{i}} = h[\boldsymbol{i} - \boldsymbol{j}],$$

and we give the following theorem.

Theorem 2 *Using the above notions, the SI metric tensor* $\mathbb{g}$ *can be decomposed as*

$$\mathbb{g}_{\boldsymbol{j}}^{\boldsymbol{i}} = \overline{\mathbb{h}}_{\boldsymbol{k}}^{\boldsymbol{i}} \mathbb{h}_{\boldsymbol{j}}^{\boldsymbol{k}},$$

where $\overline{\mathbb{h}}$ *is the tensor transpose, i.e.*

$$\overline{\mathbb{h}}_{\boldsymbol{j}}^{\boldsymbol{i}} = \mathbb{h}_{\boldsymbol{i}}^{\boldsymbol{j}}.$$

Or equivalently,

$$\langle h * \delta_{\boldsymbol{j}}, h * \delta_{\boldsymbol{i}} \rangle = \mathbb{g}_{\boldsymbol{j}}^{\boldsymbol{i}}$$

The proof is given in the appendix.

It is useful to give the following corollary of Theorem 2

$$\|\mathbb{x}\|_{\mathbb{g}}^2 = \|h * x\|^2 = \frac{1}{(2\pi)^d} \int_{T^d} \hat{g}(\boldsymbol{\omega}) |\hat{x}(\boldsymbol{\omega})|^2 \mathrm{d}\boldsymbol{\omega}, \quad (7)$$

where $T = [-\pi, \pi)$, in which $\hat{g}(\boldsymbol{\omega})$ is *filtering* the power spectrum $|\hat{x}(\boldsymbol{\omega})|^2$ that can be viewed as the *metric density*. In this sense we call $g[\boldsymbol{i}]$ a *metric filter*, which is the key in our metric learning algorithm described in the next section.

4. Learning an SI metric

In this section, we consider the metric learning problem. Based on the metric-transformation connection, we develop an metric learning method, called the *Transform Domain Metric Learning (TDML)*, by learning a *metric filter* in the transform domain.

In literature, the metric learning problem is usually formulated as optimization problem. In order to learn a metric G, one has to do optimization with respect to G. For images of size $n_1 \times n_2$, G has $n_1^2 \times n_2^2$ elements, making the optimization intractable. Another problem is that there are constraints on G which makes optimization difficult. For example, in [26], the constraint is $G \succeq 0$, so it is not easy to find efficient algorithm to solve problem with such a constraint.

In the previous sections, we have constructed an SI transform for any SI metric. Another important fact is that the function g completely describe any SI metric matrix G (or metric tensor $\mathbb{g}$). The concept of *metric filter* (7) plays the key role in our metric learning method.

We propose a novel metric learning algorithm to learn an SI metric based on the connection between metric and filter established in (7). The algorithm is efficient because the positive semi-definitive constraint

$$G \succeq 0$$

reduces to a bound constraint

$$\hat{g}(\boldsymbol{\omega}) \geqslant 0$$

and the number of parameter is the sampling number on $\hat{g}$, which is usually the same as the size of input data. Another benefit of our algorithm is that it applies to any dimensionality without modifications, thus is unnecessary to stack the multi-dimensional data to vectors.

Suppose we have some set data $\{x_i\}$, and are given the data label $\{y_i\}$. Let f_i be the Fourier transform of x_i, we compute the total "similar" and "dissimilar" power spectrum:

$$\begin{aligned} p_w(\boldsymbol{\omega}) &= \sum_{i,j,y_i=y_j} |f_i(\boldsymbol{\omega}) - f_j(\boldsymbol{\omega})|^2 \\ p_b(\boldsymbol{\omega}) &= \sum_{i,j,y_i \neq y_j} |f_i(\boldsymbol{\omega}) - f_j(\boldsymbol{\omega})|^2. \end{aligned}$$

Since

$$\frac{1}{(2\pi)^d} \int_{T^d} |f_i(\boldsymbol{\omega}) - f_j(\boldsymbol{\omega})|^2 \mathrm{d}\boldsymbol{\omega}$$

is the squared Euclidean distance between x_i and x_j, then

$$\int_{T^d} p_w(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}, \int_{T^d} p_d(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}$$

are proportional to the average within-class and between-class Euclidean distances. Similarly, with respect to a metric filter $g[\boldsymbol{i}]$,

$$\int_{T^d} \hat{g}(\boldsymbol{\omega}) p_w(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}, \int_{T^d} \hat{g}(\boldsymbol{\omega}) p_b(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}$$

are proportional to the average within-class and between-class Euclidean distance of the transformed data, respectively.

We use the criterion that the filtered within-class distance

$$\int_{T^d} \hat{g}(\boldsymbol{\omega}) p_w(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}$$

is minimized, and the filtered between-class distance

$$\int_{T^d} \hat{g}(\boldsymbol{\omega}) p_b(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}$$

is maximized, simultaneously. This gives the objective functional

$$J_0(g) = \frac{\int_{T^d} \hat{g}(\boldsymbol{\omega}) p_w(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}}{\int_{T^d} \hat{g}(\boldsymbol{\omega}) p_b(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega}}, \quad (8)$$

and the optimization problem[3]:

$$\begin{aligned} \min_{g} \quad & J_0(g) \quad (9) \\ s.t. \quad & \hat{g}(\boldsymbol{\omega}) = \chi_A(\boldsymbol{\omega}), \\ & \int_{T^d} \hat{g}(\boldsymbol{\omega}) \, \mathrm{d}\boldsymbol{\omega} = \varepsilon \cdot (2\pi)^d, \end{aligned}$$

[3]With the constraint $\hat{g}(\boldsymbol{\omega}) \geqslant 0$, the solution of the optimization problem collapses to a delta function. To ensure $\hat{g}$ does not collapse, we use the constraint $\hat{g}(\boldsymbol{\omega}) = \chi_A(\boldsymbol{\omega})$ instead.

where $\chi_A(\boldsymbol{\omega})$ is the characteristic function of set $A \subset T^d$, i.e.

$$\chi_A(\boldsymbol{\omega}) = \begin{cases} 1, & \boldsymbol{\omega} \in A \\ 0, & \text{else}. \end{cases}$$

The last constraint in (9) is a regularity condition and ε is a parameter to control the frequency coverage of the metric filter.

The solution of this optimization problem can be given in closed-form. Specifically, if the data x_i is of size $n_1 \times n_2 \times \cdots \times n_d$, we sample $n_1 \times n_2 \times \cdots \times n_d$ points of the continuous spectrum, or equivalently we replace DTFT with DFT. Then the discrete version of (9) is given as:

$$\begin{aligned} \min_{g} \quad & \tilde{J}_0(g) \\ s.t. \quad & \hat{g}[\boldsymbol{i}] = \chi_A[\boldsymbol{i}], \\ & \sum_{\boldsymbol{i}} \hat{g}[\boldsymbol{i}] = \varepsilon \cdot \prod_{k=1}^{d} n_k \end{aligned} \tag{10}$$

where

$$\tilde{J}_0(g) = \frac{\sum_{\boldsymbol{i}} \hat{g}[\boldsymbol{i}] p_w[\boldsymbol{i}]}{\sum_{\boldsymbol{i}} \hat{g}[\boldsymbol{i}] p_b[\boldsymbol{i}]}.$$

Theorem 3 *Let η be the ε-quantile of $r[\boldsymbol{i}]$, where*

$$r[\boldsymbol{i}] = \frac{p_w[\boldsymbol{i}]}{p_b[\boldsymbol{i}]},$$

the solution $\hat{g}^\star$ of the optimization problem (10) is given by

$$\hat{g}^\star[\boldsymbol{i}] = \begin{cases} 1, & \text{if } r[\boldsymbol{i}] < \eta \\ 0, & \text{else} \end{cases}.$$

The proof is given in the appendix.

We call our metric learning method the Transform Domain Metric Learning (TDML). The criterion that the transformed within-class distance is minimized and the transformed between-class distance is maximized is similar to that in Xiang *et al.*'s metric learning method (XNZ for short) [25]. The main difference is that we are looking for a *shift-invariant* metric G, while [25] is not. A positive side effect of an SI metric is the greatly simplified computational complexity, of both time and space. For example, the method in [25] involves the construction and eigendecomposition of several matrices of size $n_1 n_2 \times n_1 n_2$, while TDML needs only matrices of size $n_1 \times n_2$ and their FFT, which is apparently more computationally efficient.

5. Experiments and Discussion

In this section, we have conducted two sets of experiments. The experiments are performed on the USPS handwritten digit database and 3 face datasets (UMIST, Yale and ORL database). The images in UMIST, Yale and ORL datasets are resized to 84×69, 80×60 and 84×69, respectively[4].

		ED	IMED	GED	XNZ	TDML
USPS		94.37	**94.97**	94.72	94.07	94.72
UMIST	2	60.88	60.90	62.05	60.96	**73.92**
	4	79.68	79.68	80.78	**89.34**	**89.09**
	6	87.25	87.29	87.95	**94.77**	94.04
Yale	2	71.41	71.41	71.11	67.73	**75.26**
	3	75.21	75.13	74.92	**79.69**	**79.33**
	4	75.19	74.71	74.29	**79.71**	**79.57**
ORL	2	81.95	81.63	80.88	81.24	**84.06**
	3	89.11	88.75	88.38	**90.03**	**89.30**
	4	92.71	92.52	92.25	**94.12**	92.71

Table 1. Comparison of image metrics on various database (%). Figures in bold face are the best result or comparable according to a student t-test.

In the experiments on the UMIST, Yale and ORL face databases, we randomly select a fixed number (2, 4, 6 for UMIST; 2, 3, 4 for Yale and ORL) of images from each class as the training set, and use the remaining images for test. We repeat the process 20 times independently and the average results are calculated. The USPS database has fixed training and test sets, thus don't need repeat. The parameter ε in the metric learning algorithm (see Theorem 3), which controls the frequency coverage of the metric filter, is set to 0.4 for USPS database, and 0.1 for all other datasets.

The goal of the first set was to compare our TDML with several other metrics, including the traditional Euclidean distance (ED), IMED [21], GED [10] and XNZ [25]. The performances are evaluated in terms of recognition rate using a nearest neighbor classifier. The recognition results are shown in Table 1. TDML significantly outperforms ED, IMED and GED on the three face databases. Compared to XNZ, TDML runs $5 \sim 10$ times faster and yields a comparable performance. A notable fact is that TDML is very robust against the sample size. In the case of very small sample size (*e.g.* 2 training samples), TDML is the definite winner.

The second set of experiments was to test whether embedding the learned SI metric in an image recognition technique, *e.g.* LDA and SVM, can improve that algorithm's accuracy. Embedding an SI metric in an algorithm is simple: First, transform all images by the corresponding SI transform and, then, run the algorithm with the transformed images as inputs. Table 2 give the results of the metric when embedded to LDA and SVM. TDML usually improves the algorithm's accuracy. The results shows that TDML can

[4]This is for the computational consideration. For instance, the original image size is 160×120 the Yale database. The methods of Xiang's and LDA will involve several 19200×19200 matrices, which are too memory expensive for our workstation with only 2G RAM.

		LDA				SVM			
		ED	IMED	GED	TDML	ED	IMED	GED	TDML
USPS		94.07	94.52	**94.77**	94.42	95.37	**95.42**	95.17	95.27
UMIST	2	68.96	71.47	73.11	**75.44**	60.33	62.02	62.45	**69.53**
	4	89.47	90.30	91.36	**93.04**	88.32	87.62	88.98	**91.42**
	6	94.85	95.03	95.49	**97.41**	93.72	93.10	92.92	**95.62**
Yale	2	67.63	68.00	**69.56**	67.97	68.90	69.12	69.23	**72.30**
	3	80.79	80.75	82.08	**83.01**	81.33	80.00	80.00	**84.83**
	4	84.33	84.57	84.24	**85.17**	80.38	81.33	81.14	**83.04**
ORL	2	79.11	80.30	**81.03**	78.30	79.25	79.07	79.00	**80.38**
	3	90.07	90.89	91.18	**92.02**	**92.00**	91.14	88.79	91.64
	4	94.15	94.42	94.77	**95.17**	94.12	94.27	94.39	**95.20**

Table 2. Classification performance of the embedded metrics on various database.

greatly improve the performance of LDA and SVM.

6. Conclusion

In this paper, we have shown that every shift-invariant metric, such as IMED, is equivalent to a shift-invariant transform plus the plain Euclidean metric. The SI property is essential and necessary for images and we argue that any intuitively reasonable image metric should be shift-invariant. Based on the equivalency, we propose the metric filter to completely capture the nature of an SI metric. An efficient metric learning algorithm, called the Transform Domain Metric Learning (TDML), is next developed. TDML tries to minimize the average within-class distance and to maximize the average between-class distance, simultaneously. This is similar to the XNZ metric learning method [25] in the flavour of the criterion. Experimental results show that TDML is $5 \sim 10$ times faster than Xiang's method and offers a comparable performance. Besides, TDML is more robust in the small sample size case. A very important ability of TDML, as of IMED and GED, is that it can be easily embedded into most image recognition algorithms. That is, by feeding the transformed data to an image classification technique, the learned metric is automatically embedded. Experiments on various datasets demonstrate the effectiveness of our method.

Acknowledgment

This work was supported by NSFC (60775005, 60635030), NKBRPC (2004CB318000) and Program for New Century Excellent Talents in University.

A. Appendix

A.1. Proof of Theorem 2

$h[\boldsymbol{i}] = \mathcal{F}^{-1}\left(\sqrt{\hat{g}(\boldsymbol{\omega})}e^{\sqrt{-1}\theta(\boldsymbol{\omega})}\right)$ implies that $\hat{g}(\boldsymbol{\omega}) = |\hat{h}(\boldsymbol{\omega})|^2$. By Parseval's formula and the convolution theorem, it can be shown that

$$\begin{aligned}
\mathrm{g}_{\boldsymbol{j}}^{\boldsymbol{i}} &= g[\boldsymbol{i}-\boldsymbol{j}] \\
&= \frac{1}{(2\pi)^d}\int_{T^d}\hat{g}(\boldsymbol{\omega})\cdot e^{\sqrt{-1}(\boldsymbol{i}-\boldsymbol{j})\cdot\boldsymbol{\omega}}\mathrm{d}\boldsymbol{\omega} \\
&= \frac{1}{(2\pi)^d}\int_{T^d}|\hat{h}(\boldsymbol{\omega})|^2\cdot e^{\sqrt{-1}(\boldsymbol{i}-\boldsymbol{j})\cdot\boldsymbol{\omega}}\mathrm{d}\boldsymbol{\omega} \\
&= \frac{1}{(2\pi)^d}\left\langle \hat{h}\cdot e^{-\sqrt{-1}\boldsymbol{j}\cdot\boldsymbol{\omega}}, \hat{h}\cdot e^{-\sqrt{-1}\boldsymbol{i}\cdot\boldsymbol{\omega}}\right\rangle \\
&= \frac{1}{(2\pi)^d}\left\langle \widehat{h*\delta_{\boldsymbol{j}}}, \widehat{h*\delta_{\boldsymbol{i}}}\right\rangle \\
&= \langle h*\delta_{\boldsymbol{j}}, h*\delta_{\boldsymbol{i}}\rangle .
\end{aligned}$$

where $T = (-\pi,\pi]$, $(\boldsymbol{i}-\boldsymbol{j})\cdot\boldsymbol{\omega} = \sum_k (i_k - j_k)\omega_k$ and $\mathrm{d}\boldsymbol{\omega} = \mathrm{d}\omega_1\mathrm{d}\omega_2\cdots\mathrm{d}\omega_d$.

Because $\mathbb{h}_{\boldsymbol{j}}^{\boldsymbol{i}} = h[\boldsymbol{i}-\boldsymbol{j}]$, $\mathrm{g}_{\boldsymbol{j}}^{\boldsymbol{i}} = \langle h*\delta_{\boldsymbol{j}}, h*\delta_{\boldsymbol{i}}\rangle$ is equivalent to $\mathrm{g}_{\boldsymbol{j}}^{\boldsymbol{i}} = \overline{\mathbb{h}}_{\boldsymbol{k}}^{\boldsymbol{i}}\mathbb{h}_{\boldsymbol{j}}^{\boldsymbol{k}}$.

A.2. Proof of Theorem 3

Since η is the ε-quantile of $r[\boldsymbol{i}]$, there exist exactly $r[\boldsymbol{i}_1],\cdots,r[\boldsymbol{i}_M]$ such that $r[\boldsymbol{i}_m] < \eta$, where $m = 1,\cdots,M$ and $M = \varepsilon\cdot\prod_{k=1}^d n_k$ is fixed. The value of (10) is now

$$\tilde{J}_0 = \frac{\sum_m p_w[\boldsymbol{i}_m]}{\sum_m p_b[\boldsymbol{i}_m]}.$$

We prove that replacing any $\boldsymbol{i}_m$ will enlarge $\tilde{J}_0$, using the following inequality

$$\frac{a}{b} < \frac{a+c}{b+d} < \frac{c}{d}, \tag{11}$$

given that $\frac{a}{b} < \frac{c}{d}$, and $a,b,c,d > 0$.

By the inequality, $r[\boldsymbol{i}_m] < \eta$ implies that

$$\tilde{J}_0 = \frac{\sum_m p_w[\boldsymbol{i}_m]}{\sum_m p_b[\boldsymbol{i}_m]} < \eta. \tag{12}$$

Replacing some $\boldsymbol{i}_m$ with $\boldsymbol{i}'$, the fact $r\left[\boldsymbol{i}'\right] \geqslant \eta$ leads to

$$\frac{p_w\left[\boldsymbol{i}'\right] - \frac{1}{L}p_w\left[\boldsymbol{i}_m\right]}{p_b\left[\boldsymbol{i}'\right] - \frac{1}{L}p_b\left[\boldsymbol{i}_m\right]} \geqslant \eta, \tag{13}$$

for sufficient large $L > 0$. Combining (11), (12) and (13), we immediately have

$$\tilde{J}_0 < \tilde{J}_0'.$$

References

[1] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning Distance Functions using Equivalence Relations. In T. Fawcett and N. Mishra, editors, *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA*, pages 11–18. AAAI Press, 2003.

[2] J. Chen, R. Wang, S. Shan, X. Chen, and W. Gao. Isomap Based on the Image Euclidean Distance. In *ICPR*, pages 1110–1113. IEEE Computer Society, 2006.

[3] S. Chopra, R. Hadsell, and Y. LeCun. Learning a Similarity Metric Discriminatively, with Application to Face Verification. In *CVPR*, pages 539–546. IEEE Computer Society, 2005.

[4] A. Globerson and S. T. Roweis. Metric Learning by Collapsing Classes. In *NIPS*, 2005.

[5] J. Goldberger, S. T. Roweis, G. E. Hinton, and R. Salakhutdinov. Neighbourhood Components Analysis. In *NIPS*, 2004.

[6] R.-j. Gu, W.-b. Xu, and B. Ye. IMD-Isomap for Data Visualization and Classification. *Anti-counterfeiting, Security, Identification, 2007 IEEE International Workshop on*, pages 148–151, 2007.

[7] S. C. H. Hoi, W. Liu, M. R. Lyu, and W. ying Ma. Learning distance metrics with contextual constraints for image retrieval. In *Proc. Computer Vision and Pattern Recognition*, page 20722078. Murray Hill, 2006.

[8] D. P. Huttenlocher, G. A. Klanderman, and W. Rucklidge. Comparing Images Using the Hausdorff Distance. *IEEE Trans. Pattern Anal. Mach. Intell*, 15(9):850–863, 1993.

[9] A. K. Jain. *Fundamentals of Digital Image Processing (Prentice Hall Information and System Sciences Series)*. Prentice Hall, September 1988.

[10] J. S. N. Jean. A new distance measure for binary images. In *Acoustics, Speech, and Signal Processing, 1990. ICASSP-90., 1990 International Conference on*, pages 2061–2064, 3-6 April 1990.

[11] S. Lang. *Algebra*. Addison-Wesley, Reading, 4 edition, 1971.

[12] G. Lebanon. Metric Learning for Text Documents. *IEEE Trans. Pattern Anal. Mach. Intell*, 28(4):497–508, 2006.

[13] J. Li, G. Chen, and Z. Chi. A fuzzy image metric with application to fractal coding. *IEEE Transactions on Image Processing*, 11(6):636–643, 2002.

[14] Y. Liu. Distance Metric Learning: A Comprehensive Survey, 2006.

[15] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall Signal Processing Series. Prentice Hall, Englewood Cliffs, NJ, USA, 1989.

[16] G. Polya and G. Szego. *Aufgaben und Lehrsatze aus der Analysis*. Springer-Verlag, Berlin, 1971.

[17] J. Qiao, J. Liu, and C. Zhao. A Novel SVM-Based Blind Super-Resolution Algorithm. In *Neural Networks, 2006. IJCNN 06. International Joint Conference on*, pages 2523–2528, 16-21 July 2006.

[18] H. J. Seo, Y. K. Park, and J. K. Kim. Common Image Method(Null Space + 2DPCAs) for Face Recognition. In J. Blanc-Talon, W. Philips, D. Popescu, and P. Scheunders, editors, *Advanced Concepts for Intelligent Vision Systems, 8th International Conference, ACIVS 2006, Antwerp, Belgium, September 18-21, 2006, Proceedings*, volume 4179, pages 699–709. Springer, 2006.

[19] P. Simard, Y. L. Cun, and J. Denker. Efficient Pattern Recognition Using a New Transformation Distance. In S. J. Hanson, J. D. Cowan, and C. L. Giles, editors, *Advances in Neural Information Processing Systems*, volume 5, pages 50–58. Morgan Kaufmann, San Mateo, CA, 1993.

[20] T. Tangkuampien and D. Suter. 3D Object Pose Inference via Kernel Principal Component Analysis with Image Euclidian Distance (IMED). In *British Machine Vision Conference*, pages –137, 2006.

[21] L. Wang, Y. Zhang, and J. Feng. On the Euclidean Distance of Images. *IEEE Trans. Pattern Anal. Mach. Intell*, 27(8):1334–1339, 2005.

[22] R. Wang, J. Chen, S. Shan, and W. Gao. Enhancing Training Set for Face Detection. In *ICPR 06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 477–480, Washington, DC, USA, 2006. IEEE Computer Society.

[23] Weinberger and Saul. Fast solvers and efficient implementations for distance metric learning. In *ICML '08: Proceedings of the 25th international conference on Machine learning*, page 11601167, New York, NY, USA, 2008. ACM.

[24] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. In *NIPS*, 2005.

[25] S. Xiang, F. Nie, and C. Zhang. Learning a mahalanobis distance metric for data clustering and classification. *Pattern Recognition*, 41(12):3600 – 3612, 2008.

[26] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. J. Russell. Distance Metric Learning with Application to Clustering with Side-Information. In S. Becker, S. Thrun, and K. Obermayer, editors, *NIPS*, pages 505–512. MIT Press, 2002.

[27] S. Zhu, Z. Song, and J. Feng. Face recognition using local binary patterns with image Euclidean distance. In *MIPPR 2007: Remote Sensing and GIS Data Processing and Applications; and Innovative Multispectral Technology and Applications. Edited by Wang, Yongji; Li, Jun; Lei, Bangjun; Yang, Jingyu. Proceedings of the SPIE, Volume 6790, pp. 67904Z (2007).*, volume 6790, nov 2007.