Multi-Manifold Deep Metric Learning for Image Set Classification

Jiwen Lu[1], Gang Wang[1,2], Weihong Deng[3], Pierre Moulin[1,4], and Jie Zhou[5]
[1]Advanced Digital Sciences Center, Singapore
[2]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
[3]School of ICE, Beijing University of Posts and Telecommunications, Beijing, China
[4]Department of ECE, University of Illinois at Urbana-Champaign, Urbana, IL, USA
[5]Department of Automation, Tsinghua University, Beijing, China
jiwen.lu@adsc.com.sg; wanggang@ntu.edu.sg; whdeng@bupt.edu.cn;
moulin@ifp.uiuc.edu; jzhou@tsinghua.edu.cn

Abstract

In this paper, we propose a multi-manifold deep metric learning (MMDML) method for image set classification, which aims to recognize an object of interest from a set of image instances captured from varying viewpoints or under varying illuminations. Motivated by the fact that manifold can be effectively used to model the nonlinearity of samples in each image set and deep learning has demonstrated superb capability to model the nonlinearity of samples, we propose a MMDML method to learn multiple sets of nonlinear transformations, one set for each object class, to nonlinearly map multiple sets of image instances into a shared feature subspace, under which the manifold margin of different class is maximized, so that both discriminative and class-specific information can be exploited, simultaneously. Our method achieves the state-of-the-art performance on five widely used datasets.

1. Introduction

Image set classification has been an important problem in computer vision in recent years [1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13, 15, 16, 18, 22, 24, 26, 27, 28, 29, 34, 35, 36, 37, 38, 39, 40], especially when more and more data are easily accessible and multiple images of the same object are easily captured nowadays. There are many practical applications for image set classification such as visual surveillance, multi-view camera network analysis, and personal album organization. Generally, image set classification aims to recognize an object of interest from a set of image instances captured from varying viewpoints or under varying illuminations, which is different from the conventional image classification where each training and testing example is a single still image. Compared to a single image, an image set offers us more useful information to describe objects of interest. However, it is also more challenging to exploit

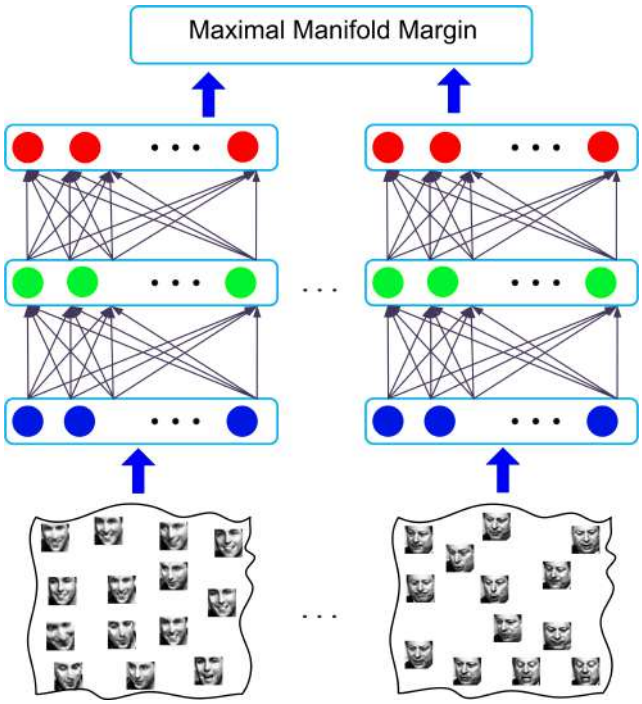


Figure 1. The basic idea of our proposed image set classification approach. For each image set, we model it as a manifold and pass it into multiple layers of deep neural networks to nonlinearly map each manifold into another feature space. Specifically, the deep network is class-specific so that different classes have different parameters in their networks. In the top layer of these networks, the maximal manifold margin criterion is used to learn the parameters of these manifold. In the testing stage, we apply these class-specific deep networks to compute the similarity between the testing image set and all training classes and the smallest distance is used for classification.

discriminative information from image sets because there are usually larger intra-class variations within a set, which makes the classification task more difficult.

There have been a variety of studies on image set classification over the past decade [1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13, 15, 16, 18, 22, 24, 26, 27, 28, 29, 34, 35, 36, 37, 38, 39, 40], and significant progresses have been made in recent years [3, 4, 5, 11, 26, 27, 35, 37, 39, 40]. One key challenge in image set classification is how to effectively model and represent each image set because there are usually high nonlinearity of samples within a set. While existing methods have achieved reasonably good performance in image set classification, most of them usually make strong assumptions such as singe Gaussian, Gaussian mixture models, subspace or mixture of subspaces to represent image sets. In many real world applications, these assumptions may not be held, especially when there are complex variations within a set.

In this paper, we propose a new multi-manifold deep metric learning (MMDML) approach for image set classification, where the key idea of the proposed approach is shown in Figure 1. Given each image set, we first model it as a nonlinear manifold because manifolds can effectively describe the geometrical and structural information of image instances within image sets. Motivated by the fact that deep learning has demonstrated superb capability to model the nonlinearity of samples, we propose a MMDML method to learn multiple sets of nonlinear transformations, one set for each object class, to nonlinearly map multiple sets of image instances into a shared feature subspace, under which the manifold margin of different class is maximized, so that both discriminative and class-specific information can be exploited, simultaneously. Experimental results on five widely used datasets validate the effectiveness of the proposed method.

2. Related Work

Image Set Classification: Existing image set classification methods [1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13, 15, 16, 18, 22, 24, 26, 27, 28, 29, 34, 35, 36, 37, 38, 39, 40] can be categorized into two classes: parametric and non-parametric. For the first category, each image set is modeled as a specific distribution and then the Kullback-Leibler (KL) divergence is used to compute the similarity of two image sets. For example, Shakhnarovich *et al.* [28] modeled each image set as a single Gaussian [28], Arandjelovic *et al.* [1] represented each image set as a Gaussian mixture model. The key limitation of this class of methods is that if there is no strong correlation between two image sets, such a parametric model cannot well characterize the image sets and hence the similarity estimated is not effective. For the second category, each image set is modeled as a subspace [13, 39], covariance descriptor [27, 35], affine or convex hull [2] or dictionary [5, 26]. Then, the distance between these nonparametric models is utilized to compute the similarity of two image sets. However, most of these nonparametric methods are linear models, which are generally not strong enough to model image sets, especially when there are complex variations within a set. To address this, Hayat *et al.* [11] presented a deep learning approach for image set classification, where multiple layers of non-linear reconstruction models were used to model image set. While encouraging performance was achieved, their approach is generative, which is not discriminative enough to differentiate different objects. In this work, we propose a discriminative deep learning approach to extract more discriminative information for image set classification, and we achieve superior or very competitive results on five widely used datasets.

Deep Learning: Recently, deep learning has attracted increasing interest in computer vision and machine learning, and a variety of deep learning algorithms have been proposed over the past few years [12, 14, 17, 20, 21]. Generally, deep learning aims to build high-level features by learning hierarchical feature representations from raw data. Representative deep learning models included deep stacked auto-encoder [20], deep convolutional neural networks [40], and deep belief network [12], and some of them have been successfully employed in various vision applications such as image classification [17], object detection [30], action recognition [20], face verification [31], and visual tracking [33]. While significant progress has been achieved, little attempt has been made on deep learning for image set classification. To our knowledge, [11] is the first work on using deep learning for image set classification, where person-specific nonlinear deep reconstruction models are learned for classification. However, their method is unsupervised, which may not be discriminative enough to extract nonlinear information for classification. In this work, we propose a discriminative deep learning method to exploit both the nonlinear and discriminative information for image set classification.

3. Proposed Approach

Figure 1 shows the basic idea of our proposed MMDML method, and the following subsections present the details of the proposed method.

3.1. MMDML

Let $X = [X_1, \cdots, X_c, \cdots, X_C]$ be the training set of C different classes, where $X_c = [x_{c1}, x_{c2}, \cdots, x_{ci}, \cdots, x_{cN_c}] \in \mathbb{R}^{d \times N_c}$ denotes the cth image set, $1 \leq c \leq C$, N_c is the number of samples in this image set[1], x_{ci} is the ith image in this image

[1] In the training set, there could be multiple image sets for some classes. For this case, we merge image sets from the same person into a large image set to learn the class-specific deep model.

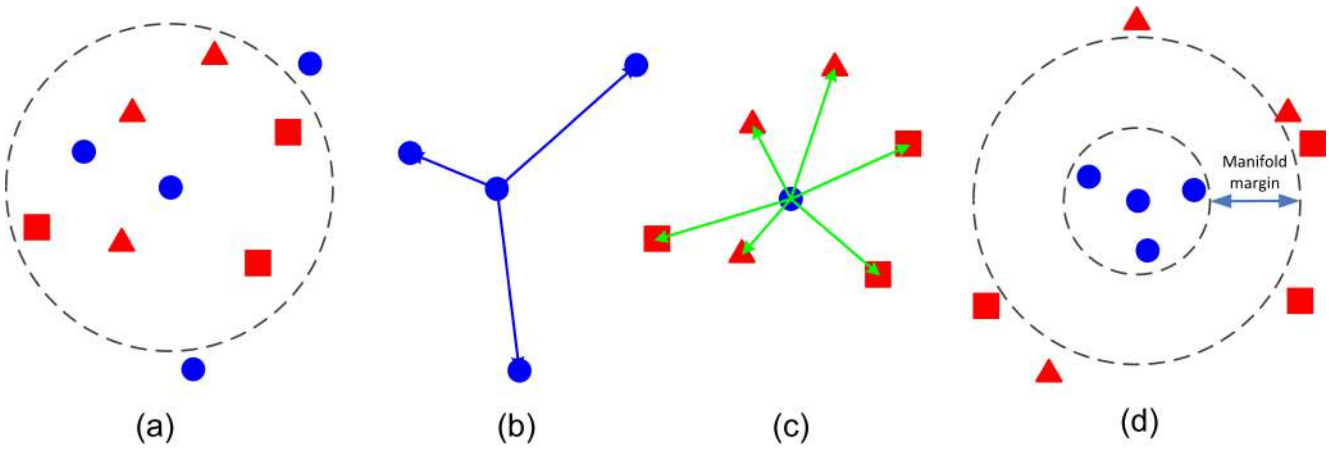


Figure 2. Illustration of the maximal manifold margin criterion used in our MMDML method. (a) There are thee intra-manifold neighbors (denoted as the same circles) and six inter-manifold neighbors (denoted as the squares and triangles). (b) The intra-manifold neighbors. (c) The inter-manifold neighbors. (d) The manifold margin is maximized after applying our MMDML method.

set, and d is the feature dimension of each image. As shown in Figure 1, we construct a deep neural network for each class, and pass the image set X_c into the cth network. Assume there are $L+1$ layer in the work, and d_c^l denote the number of nodes in lth layer of the cth network, where $1 \leq l \leq L$. For the image x_{ci}, its output of the first layer in the cth network is computed as: $h_{ci}^1 = s(W_c^1 x_{ci} + b_c^1)$, where W_c^1 is the projection matrix and b_c^1 is the bias vector to be learned in the first layer of the cth network, s is a nonlinear active function which applies component-wisely, which is widely used in previous deep learning algorithms [12, 14, 17, 20, 21]. Then, the output of the first layer of this network is used as the input of the second layer. Therefore, the output of the second layer is $h_{ci}^2 = s(W_c^2 h_{ci}^1 + b_c^2)$, where W_c^2 is the projection matrix and b_c^2 is the bias vector to be learned in the second layer of the cth network, respectively. Similarly, the output for the lth layer is $h_{ci}^l = s(W_c^l h_{ci}^{l-1} + b_c^l)$, and for the top layer is:

$$h_{ci}^L = s(W_c^L h_{ci}^{L-1} + b_c^L) \tag{1}$$

where W_c^L is the projection matrix and b_c^L is the bias vector to be learned for the top layer of the cth network, respectively.

To boost the image set classification performance, we expect that image sets from different classes can be well separated at the top layer of the learned deep networks. Since each image set is modeled as a manifold, we aim to maximize the margin of different manifolds from different classes so that discriminative information is extracted for classification. While there have been several works on computing the manifold-manifold distance [25, 34, 36], there is still a lack of a formal definition of manifold-manifold distance. In our work, for each sample h_{ci}^L from the cth manifold, we compute two squared distances $D_1(h_{ci}^L)$ and $D_2(h_{ci}^L)$, which measure the dissimilarity between this sample and its intra-class and inter-class neighbors as follows:

$$D_1(h_{ci}^L) = \frac{1}{K_1}\sum_{p=1}^{K_1} \|h_{ci}^L - h_{cip}^L\|_2^2 \tag{2}$$

$$D_2(h_{ci}^L) = \frac{1}{K_2}\sum_{q=1}^{K_2} \|h_{ci}^L - h_{ciq}^L\|_2^2 \tag{3}$$

where h_{cip}^L and h_{ciq}^L are the feature representations at the top layer of the pth intra-manifold and qth inter-manifold neighbors, K_1 and K_2 are two parameters to define the neighborhood size, respectively.

Let $f_c = \{W_c^1, W_c^2, \cdots, W_c^L, b_c^1, b_c^2, \cdots, b_c^L\}$ be the parameters of the cth network, we formulate the following optimization problem to maximize the margin between the cth manifold and other manifolds:

$$\min_{f_c} \sum_{i=1}^{N_c} \left(D_1(h_{ci}^L) - D_2(h_{ci}^L)\right) \tag{4}$$

The objective in (4) is to ensure that for each face sample x_{ci} from the cth class, the distance between it and the K_1 intra-manifold neighbors is minimized and that between it and the K_2 inter-manifold neighbors is maximized, so that large margin can be exploited for each sample in this manifold. Figure 2 presents an illustration to show the key idea of how these intra-manifold and inter-manifold neighbors are constrained to maximize the manifold margin, where K_1 and K_2 are set as 3 and 6, respectively.

By applying the criterion in (4) on each sample from all image sets in the training set, we formulate the following

optimization problem for our MMDL modal:

$$\begin{aligned}\min_{f_1,f_2,\cdots,f_C} H &= H_1 + \frac{\lambda}{2}H_2 \\ &= \sum_{c=1}^{C}\sum_{i=1}^{N_c} g\left(D_1(h_{ci}^L) - D_2(h_{ci}^L)\right) \\ &+ \frac{\lambda}{2}\sum_{c=1}^{C}\sum_{l=1}^{L}\left(\|W_c^l\|_F^2 + \|b_c^l\|_2^2\right) \quad (5)\end{aligned}$$

where H_1 maximizes the manifold margins to exploit the discriminative information for classification, and H_2 regularizes the parameters of these networks, λ is a parameter to balance the contributions of different terms, and $g(a)$ is a generalized logistic loss function to smoothly approximate the hinge loss function $a = \max(a, 0)$, and is defined as follows:

$$g(a) = \frac{1}{\rho}\log(1 + \exp(\rho a)) \quad (6)$$

where ρ is the sharpness parameter.

Since h_{cip}^L and h_{ciq}^L depend on the network parameters $W_c^1, W_c^2, \cdots, W_c^L$, and $b_c^1, b_c^2, \cdots, b_c^L$, which are also to be learned in our method, the optimization function defined in (5) is an egg and chicken problem. To address this, we develop an iterative algorithm to obtain a local optimal solution. Specifically, we first initialize the network parameters with appropriate values and compute the intra-class and inter-class neighbors, then, we update these parameters by (5) until convergence.

We adopt the stochastic sub-gradient descent algorithm to solve the optimization problem in (5) to obtain the parameters $\{W_c^1, b_c^l\}|_{l=1}^L$. The gradient of the objective function H with respect to W_c^1 and b_c^l can be computed as follows:

$$\begin{aligned}\frac{\partial H}{\partial W_c^l} &= \sum_{i=1}^{N_c}\left(\delta_{ci}^l(h_{ci}^{l-1})' + \delta_{cip}^l(h_{cip}^{l-1})' + \delta_{ciq}^l(h_{ciq}^{l-1})'\right) \\ &\quad +\lambda W_c^l \quad (7) \\ \frac{\partial H}{\partial b_c^l} &= \sum_{i=1}^{N_c}\left(\delta_{ci}^l + \delta_{cip}^l + \delta_{ciq}^l\right) + \lambda b_c^l \quad (8)\end{aligned}$$

where δ_{ci}^l, δ_{cip}^l and δ_{ciq}^l are three updating functions. For the top layer ($l = L$), they are computed as follows:

$$\begin{aligned}\delta_{ci}^l &= g'(D)(R_1 + R_2) \odot s'(y_{ci}^L) \quad (9) \\ \delta_{cip}^l &= -g'(D)R_1 \odot s'(y_{cip}^L) \quad (10) \\ \delta_{ciq}^l &= -g'(D)R_2 \odot s'(y_{ciq}^L) \quad (11)\end{aligned}$$

where

$$\begin{aligned}D &\triangleq D_1(h_{ci}^L) - D_2(h_{ci}^L) \quad (12) \\ R_1 &\triangleq \frac{1}{K_1}\sum_{p=1}^{K_1}(h_{ci}^L - h_{cip}^L) \quad (13) \\ R_2 &\triangleq \frac{1}{K_2}\sum_{p=1}^{K_2}(h_{ci}^L - h_{ciq}^L) \quad (14) \\ y_{ci}^l &\triangleq W_c^l h_{ci}^{l-1} + b_c^l \quad (15) \\ y_{cip}^l &\triangleq W_c^l h_{cip}^{l-1} + b_c^l \quad (16) \\ y_{ciq}^l &\triangleq W_c^l h_{ciq}^{l-1} + b_c^l \quad (17)\end{aligned}$$

For all other layers, $1 \le l \le L-1$, δ_{ci}^l, δ_{cip}^l and δ_{ciq}^l are computed as follows:

$$\begin{aligned}\delta_{ci}^l &= (W_c^{l+1})^T\delta_{ci}^{l+1} \odot s'(y_{ci}^l) \quad (18) \\ \delta_{cip}^l &= (W_c^{l+1})^T\delta_{cip}^{l+1} \odot s'(y_{cip}^l) \quad (19) \\ \delta_{ciq}^l &= (W_c^{l+1})^T\delta_{ciq}^{l+1} \odot s'(y_{ciq}^l) \quad (20)\end{aligned}$$

where the operation "$\odot$" denotes the element-wise multiplication.

Then, we use the the following gradient descent algorithm to update the parameters W_c^l and b_c^l of our networks:

$$\begin{aligned}W_c^l &= W_c^l - \mu\frac{\partial H}{\partial W_c^l} \quad (21) \\ b_u^l &= b_u^l - \mu\frac{\partial H}{\partial b_u^l} \quad (22)\end{aligned}$$

where μ is the learning rate, $1 \le c \le C$, $1 \le l \le L$.

The proposed MMDML method is summarized in **Algorithm 1**.

3.2. Classification

Given a testing image set $X^q = [x_1^q, x_2^q, \cdots, x_{N_q}^q]$, where x_j^q is the jth image ($1 \le j \le N_q$) in this set and N_q is the number of images in this set, we compute the distance between the testing set X_q and each training set X_c, and assign a label L_q to the testing image set X_q as follows:

$$L_q = \arg\min_c d(X_q, X_c), \quad 1 \le c \le C. \quad (23)$$

Now, we discuss how to compute the distance $d(X_q, X_c)$ in our experiments. For each sample x_j^q, we first use the learned deep network from the cth class to map it into the feature space $h_c(x_j^q)$. Then, we compute the distance between $h_c(x_j^q)$ and each training sample h_{ci} in the feature space from the cth manifold by using the Euclidean distance, then the smallest distance between $h_c(x_j^q)$ and h_{ci} is selected as the distance between x_j^q and the cth manifold. Finally, we average all these point-to-manifold distance as the distance between manifold X^q and X^c.

Algorithm 1: MMDML

Input: Training set X, network layer number $L+1$, learning rate μ, iterative number T, parameter λ, K_1 and K_2, and convergence error ε.

Output: Parameters W_c^l and b_c^l, $1 \leq c \leq C$, $1 \leq l \leq L$.

Step 1 (Initialization):

Initialize W_c^l and b_c^l with appropriate values.

Step 2 (Optimization by back prorogation):

for $t = 1, 2, \cdots, T$ **do**

  Compute the intra-manifold and inter-manifold neighbors.

  for $l = 1, 2, \cdots, L$ **do**

    Compute h_{ci}^l, h_{cip}^l, and h_{ciq}^l using the deep networks.

  end

  for $l = L, L-1, \cdots, 1$ **do**

    Obtain the gradients according to (7)-(8).

  end

  for $l = 1, 2, \cdots, L$ **do**

    Update W_u^l, W_v^l, b_u^l and b_v^l according to (21)-(22).

  end

  Calculate H_t using (5).

  If $t > 1$ and $|H_t - H_{t-1}| < \varepsilon$, go to **Return**.

end

Return: W_c^l and b_c^l, where $1 \leq c \leq C$, $1 \leq l \leq L$.

3.3. Discussion

Both [11] and our approach are deep learning based image set matching methods. The key difference is that our model is supervised while theirs is unsupervised. Hence, our method requires more labeled examples to learn the model because more parameters to be estimated in our method.

4. Experimental Results

We conducted image set classification experiments on five publicly available datasets including the Honda/UCSD [22], CMU Mobo [9], YouTube Celebrities (YTC) [15], PubFig [19] face datasets, and the ETH-80 object dataset [23]. We describe the details of the experiments and results in the following.

4.1. Datasets

The Honda/UCSD dataset [22] contains 59 face video sequences of 20 different persons. The number of frames for these video varies from 12 to 645. There are large variations in facial expression and head pose in this dataset.

The Mobo dataset [9] was originally created for gait recognition. There are 96 video sequences of 24 different persons, and each person contains 4 videos captured from different walking conditions, such as slow walking, fast walking, inclined walking, and walking with a ball. For each video, there are around 300 frames covering variations of pose and expressions.

The YTC dataset [15] contains 1910 face video sequences of 47 different persons, who are celebrities such as actors, actresses and politicians. Face videos in this dataset were collected from YouTube under unconstrained conditions. There are large variations of pose, illumination, and expression on face videos in this dataset. Moreover, the quality of face videos is very poor because most videos are of high compression rate. The number of frames for face videos varies from 7 to 400.

The PubFig dataset [19] contains 58797 images of 200 different persons. There are large variations of pose, illumination, expression on face images because these real-life face images were captured in unconstrained environments from the internet.

The ETH-80 dataset [23] contains visual object images from 8 different categories including apples, cars, cows, cups, dogs, horses, pears and tomatoes. For each category, there are 10 object instances and 41 images for each object instance captured from different viewpoints.

4.2. Experimental Settings

For face videos in the Honda, Mobo and YTC datasets, we employed the face detector presented in [32] to detect each face image frame and then resized it into 20×20. For face images in the PubFig dataset, we cropped face region of each face image according to the provided bounding box position, and resized it into 20×20. We applied histogram equalization on each image from all these four face datasets to remove the illumination effect. For the ETH-80 dataset, each object image was segmented from the simple background and scaled to 20×20 for classification, which is consistent to previous studies in [11, 27, 35]. Finally, each image in all the five datasets was lexicographically into a 400-dimensional feature vector. Unlike face recognition, the task on ETH-80 is to classify each image set of an object into a pre-defined category.

For the Honda, Mobo and YTC datasets, image frames extracted from each face video were considered as an image set. For the PubFig dataset, we equally divided face images of each person into three folds, where three different image sets were constructed for evaluation. On the Honda and Mobo datasets, we conducted experiments 10 times by randomly selecting different training and testing sets. For the YTC dataset, we employed the five fold cross validation strategy by following the same setting in [11, 26, 27, 34, 35]. Specifically, we equally divided the whole dataset into five folds (with minimal overlapping), where each fold contains 9 different images for each person. For each fold, 3 image sets were randomly selected for training and the rest 6 were used for testing. For the PubFig dataset, we used one fold for training and the remaining two for testing by random-

ly selecting different folds for training and testing. For the ETH-80 dataset, we randomly selected 5 objects from each category for training and the remaining 5 for testing. On all the five datasets, the average classification rate and the standard deviation were used to evaluate different image set classification methods.

For our MMDML method, we designed our deep model with two layers, and the feature dimensions for these layers were set as 400, 200, and 100, respectively. The learning rate μ, parameter λ, $K1$ and K_2 were empirically set as 0.0001, 0.00001, 5 and 20, respectively[2]. The parameters W_u^l and W_v^l of our CDML model were initialized as $\mathbf{E} \in \mathbb{R}^{d_c^l \times d_c^{l-1}}$ (d_c^l is the feature dimension of the lth layer), which is a matrix with ones on the diagonal and zeros elsewhere. The bias vector b_c^l was initialized as zero vectors. For the active function, we used the non-saturating sigmoid function in our experiments.

4.3. Results and Analysis

Comparison with State-of-the-Art Image Set Classification Methods: We first compared our MMDML method with twelve state-of-the-art image set classification methods, including Mutual Subspace Method (MSM) [38], Discriminant Canonical Correlation analysis (DCC) [16], Manifold-to-Manifold Distance (MMD) [36], Manifold Discriminant Analysis (MDA) [34], Affine Hull based Image Set Distance (AHISD) [2], Convex Hull based Image Set Distance (CHISD) [2], Sparse Approximated Nearest Point (SANP) [13], Covariance Discriminative Learning (CDL) [35], Dictionary-based Face Recognition from Video (DFRV) [5], Local Multi-Kernel Metric Learning (LMKML) [27], Set-to-Set Distance Metric Learning (SSDML) [40], and Simultaneous Feature and Dictionary Learning (SFDL) [26]. We employed the implementations of these compared methods provided by the original authors except the DFRV method because the code of DFRV was not publicly available. We implemented the DFRV method by following the algorithm description in [5]. For all these twelve compared methods, we used the default parameters recommended by the corresponding papers. For the DCC, MDA, CDL and LMKML methods, if there is a single image set from each class on the Honda, Mobo, and PubFig datasets, we randomly and equally divided each training image set into two subsets for discriminative learning, so that the intra-class variation can be effectively modeled.

Table 1 tabulates the average classification rates and standard deviations of different image set classification methods on all the five datasets. We clearly see that our MMDML method achieves higher classification rate than all the other compared state-of-the-art methods on all the five datasets. Compared to those unsupervised image set classification methods such as MSM, DCC, MMD, AHISD, CHISD, SANP, and DFRV, our MMDML can extract discriminative information in the learned deep networks. Compared to those supervised image set classification methods such as MDA, CDL, LMKML, SDDML, and SFDL, our MMDML is a deep learning approach which explicitly addresses the nonlinear separation problem by learning multiple sets of nonlinear transformations, so that more discriminative, nonlinear, and class-specific information can be exploited to improve the classification performance.

[2]We tuned these parameters by using the 5-fold cross-validation strategy on the training set of the YTC dataset.

Table 2. Average classification rates and the standard deviations (%) of multi-manifold deep learning and multi-manifold shallow learning methods on different datasets.

Method	MMSML	MMKML	MMDML
Honda	95.5 ± 0.9	97.5 ± 0.4	$\mathbf{100.0 \pm 0.0}$
Mobo	94.5 ± 1.7	96.5 ± 1.4	$\mathbf{97.8 \pm 1.0}$
YTC	74.5 ± 3.5	76.7 ± 3.4	$\mathbf{78.5 \pm 2.8}$
PubFig	75.5 ± 2.4	80.4 ± 1.8	$\mathbf{82.5 \pm 1.2}$
ETH-80	90.5 ± 4.5	91.7 ± 4.3	$\mathbf{94.5 \pm 3.5}$

Comparison with Different Multi-Manifold Learning Strategies: We compared our MMDML with two other different multi-manifold learning strategies:

1. Multi-Manifold Shallow Metric Learning (MMSML): We constructed the MMSML method by setting the layer of each network to one and determining the active function $s(z) = z$ in our MMDML.

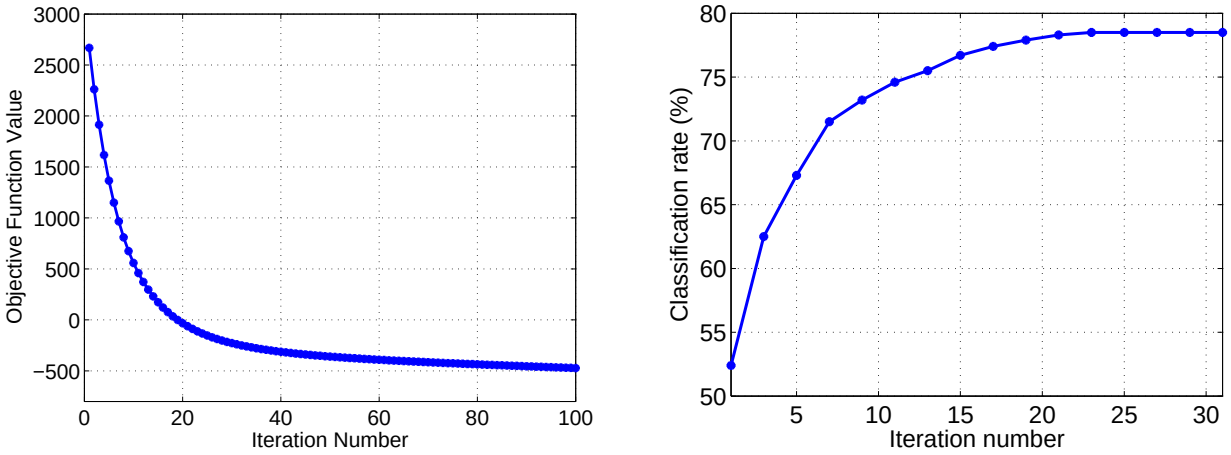
2. Multi-Manifold Kernel Metric Learning (MMKML): We employed the kernel trick on the MMSL method to the MMKML method by mapping each sample into a high-dimensional feature space. Then, we performed MMSML in the kernel space, where the RBF kernel and the average of the distance over all pairs of samples was used for evaluation.

Table 2 shows the average classification rates and standard deviations of these three different multi-manifold learning methods on different datasets. We see that our MMDML consistently outperforms MMSML and MMKML on all datasets. Compared to MMSML, our MMDML method can learn multiple hierarchical nonlinear transformations while the corresponding MMSML only learns multiple linear transformations, so that MMDML can discover the nonlinear relationship of image sets in the learned feature space. Compared to MMKML, our MMDML can explicitly seek the nonlinear mapping for each image, so that it can better describe the nonlinearity of samples to yield better classification performance.

Convergence Analysis: We evaluated the convergence of our MMDML versus different number of iterations. Figure 3(a) plots the value of the objective function of MMDML versus different number of iterations on the YTC dataset. We see that our MMDML converges in about 40 iterations.

Table 1. Average classification rates and the standard deviations (%) of different image set classification methods on different datasets.

Method	Honda	Mobo	YTC	PubFig	ETH-80	Year
MSM [38]	92.5 ± 2.3	96.5 ± 2.0	61.7 ± 4.3	57.4 ± 1.7	75.5 ± 4.9	1998
DCC [16]	92.6 ± 2.5	88.9 ± 2.5	65.8 ± 4.5	45.5 ± 1.5	91.8 ± 3.7	2006
MMD [36]	92.1 ± 2.3	92.5 ± 2.9	67.7 ± 3.8	46.3 ± 1.5	86.5 ± 4.5	2008
MDA [34]	94.5 ± 3.2	94.4 ± 2.5	68.1 ± 4.3	48.6 ± 1.6	89.2 ± 3.7	2009
AHISD [2]	91.5 ± 1.8	94.1 ± 1.5	66.5 ± 4.5	62.1 ± 1.4	78.6 ± 4.7	2010
CHISD [2]	93.7 ± 1.9	95.8 ± 1.3	67.4 ± 4.7	64.5 ± 1.5	79.7 ± 4.3	2010
SANP [13]	95.3 ± 3.1	96.1 ± 1.5	68.3 ± 5.2	78.5 ± 1.4	80.5 ± 4.7	2011
CDL [35]	97.4 ± 1.3	92.5 ± 2.9	69.7 ± 4.5	65.5 ± 1.5	86.5 ± 3.7	2012
DFRV [5]	97.4 ± 1.9	94.4 ± 2.3	74.5 ± 4.5	74.5 ± 1.4	87.5 ± 2.7	2012
LMKML [27]	98.5 ± 2.5	94.5 ± 2.5	75.2 ± 3.9	72.5 ± 1.5	92.5 ± 4.5	2013
SSDML [40]	93.5 ± 2.8	95.1 ± 2.2	74.3 ± 4.5	65.5 ± 1.7	87.5 ± 4.7	2013
SFDL [26]	98.5 ± 1.5	96.5 ± 2.3	75.7 ± 3.4	78.5 ± 1.7	90.5 ± 4.7	2014
MMDML	$\mathbf{100.0 \pm 0.0}$	$\mathbf{97.8 \pm 1.0}$	$\mathbf{78.5 \pm 2.8}$	$\mathbf{82.5 \pm 1.2}$	$\mathbf{94.5 \pm 3.5}$	



(a) (b)

Figure 3. (a) Convergence curve of MMDML on the YTC dataset. (b) Average classification rate versus different number of iterations of MMDML on the YTC dataset.

We also computed the classification rate of MMDML versus different number of iterations on the YTC dataset. Figure 3(b) shows the average classification rate of MMDML versus different number of iterations on the YTC dataset. We see that our MMDML achieves stable performance in $20 \sim 25$ iterations.

Robustness Analysis: We examined the performance of our MMDML when each image set contains different number of image samples. We randomly selected P frames from each image set and used them for model learning and classification. If one image set contains less than P image samples, all images in this set were used for classification. Table 3 shows the average classification rates of different image set classification methods on the YTC dataset, where different number of samples per set were used for evaluation. We see that the classification rate of our MMDML drops less than other compared image set classification methods. That is because in our MMDML method, the average point-manifold distance is considered as the manifold margin so that the performance of the approach depends less on the number of image samples per set than other methods such as MDA and MMD, which usually require enough samples to model the set as a nonlinear manifold. Hence, our method is not sensitive to the number of samples per set.

Computational Time: Lastly, we compared the computational time of different image set classification methods on the YTC dataset. For the test stage, we computed the classification time of classifying one image set with all training image sets. Our hardware configuration is a 2.8-GHz CPU and a 24GB RAM. Table 4 shows the time spent on the train and test stages by different image set classification methods with the Matlab software. We see that the computational time of of our MMDML in the training stage is generally higher than those of many existing methods and the testing time is comparable to those of most existing methods.

Table 3. Average classification rates and the standard deviations (%) of different image set classification methods with different number of images per set on the YTC dataset.

Method	50 frames	100 frames	All frames
MSM [38]	57.6 ± 4.5	59.4 ± 4.7	61.7 ± 4.3
DCC [16]	59.6 ± 4.8	62.6 ± 4.3	65.8 ± 4.5
MMD [36]	61.2 ± 4.2	63.9 ± 4.4	67.7 ± 3.8
MDA [34]	62.1 ± 4.6	64.4 ± 4.7	68.1 ± 4.3
AHISD [2]	60.3 ± 4.6	63.5 ± 4.9	66.5 ± 4.5
CHISD [2]	61.2 ± 4.3	64.6 ± 4.8	67.4 ± 4.7
SANP [13]	63.3 ± 5.4	65.6 ± 5.7	68.3 ± 5.2
CDL [35]	65.3 ± 4.3	67.7 ± 4.7	69.7 ± 4.5
DFRV [5]	70.5 ± 4.7	72.5 ± 4.4	74.5 ± 4.5
LMKML [27]	71.2 ± 4.4	73.2 ± 3.7	75.2 ± 3.9
SSDML [40]	69.5 ± 4.7	72.3 ± 4.2	74.3 ± 4.5
SFDL [26]	72.3 ± 3.7	74.4 ± 3.4	75.7 ± 3.4
MMDML	$\mathbf{75.5 \pm 2.4}$	$\mathbf{76.7 \pm 2.6}$	$\mathbf{78.5 \pm 2.8}$

Table 4. Computation time (seconds) of different image set classification methods on the YTC dataset for training and testing (classification of one image set.

Method	Train	Test	Method	Train	Test
MSM	N.A	2.7	DCC	97.9	2.5
MMD	N.A	3.5	MDA	178.5	3.2
AHISD	N.A	8.4	CHISD	N.A	6.7
SANP	N.A	45.6	CDL	67.9	12.6
DFRV	8660.2	5.2	LMKML	4228.5	5.2
SSDML	23.3	2.5	SFDL	7545.3	6.4
MMDML	1534.3	5.4			

5. Conclusion and Future Work

In this paper, we have proposed a multi-manifold deep learning (MMDML) method for image set classification. By jointly learning multiple sets of nonlinear transformations (one set for each class), our method nonlinearly maps multiple sets of image instances into a shared feature subspace, so that discriminative, class-specific and nonlinear information are exploited for classification. Experimental results on five popular datasets have demonstrated that our method achieves better performance than the state-of-the-art image set classification methods.

For future work, we are interested in applying our proposed method to other vision applications such as image set based person re-identification and video-based action recognition to further demonstrate its effectiveness.

Acknowledgement

This work was supported by a research grant from the Agency for Science, Technology and Research of Singapore for the Human Cyber Security Systems (HCSS) Program at the Advanced Digital Sciences Center, the research grant of Singapore Ministry of Education (MOE) Tier 2 ARC28/14, Singapore A*STAR Science and Engineering Research Council PSF1321202099, and the National Natural Science Foundation of China under Grant 61225008, the National Basic Research Program of China under Grant 2014CB349304, the Ministry of Education of China under Grant 20120002110033, and the Tsinghua University Initiative Scientific Research Program.

References

[1] O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell. Face recognition with image sets using manifold density divergence. In *CVPR*, pages 581–588, 2005. 1, 2

[2] H. Cevikalp and B. Triggs. Face recognition based on image sets. In *CVPR*, pages 2567–2573, 2010. 1, 2, 6, 7, 8

[3] L. Chen. Dual linear regression based classification for face cluster recognition. In *CVPR*, pages 2673–2680, 2014. 1, 2

[4] S. Chen, C. Sanderson, M. T. Harandi, and B. C. Lovell. Improved image set classification via joint sparse approximated nearest subspaces. In *CVPR*, pages 452–459, 2013. 1, 2

[5] Y.-C. Chen, V. M. Patel, P. J. Phillips, and R. Chellappa. Dictionary-based face recognition from video. In *ECCV*, pages 766–779, 2012. 1, 2, 6, 7, 8

[6] Z. Cui, S. Shan, H. Zhang, S. Lao, and X. Chen. Image sets alignment for video-based face recognition. In *CVPR*, pages 2626–2633, 2012. 1, 2

[7] W. Fan and D. Yeung. Locally linear models on face appearance manifolds with application to dual-subspace based classification. In *CVPR*, pages 1384–1390, 2006. 1, 2

[8] A. Fitzgibbon and A. Zisserman. Joint manifold distance: a new approach to appearance based clustering. In *CVPR*, pages 26–33, 2003. 1, 2

[9] R. Gross and J. Shi. The cmu motion of body (mobo) database. Technical report, Carnegie Mellon University, 2001. 5

[10] M. Harandi, C. Sanderson, S. Shirazi, and B. Lovell. Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching. In *CVPR*, pages 2705–2712, 2011. 1, 2

[11] M. Hayat, M. Bennamoun, and S. An. Learning non-linear reconstruction models for image set classification. In *CVPR*, pages 1915–1922, 2014. 1, 2, 5

[12] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006. 2, 3

[13] Y. Hu, A. Mian, and R. Owens. Sparse approximated nearest points for image set classification. In *CVPR*, pages 121–128, 2011. 1, 2, 6, 7, 8

[14] G. B. Huang, H. Lee, and E. Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In *CVPR*, pages 2518–2525, 2012. 2, 3

[15] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley. Face tracking and recognition with visual constraints in real-world videos. In *CVPR*, pages 1–8, 2008. 1, 2, 5

[16] T. Kim, J. Kittler, and R. Cipolla. Learning discriminative canonical correlations for object recognition with image sets. In *ECCV*, pages 251–262, 2006. 1, 2, 6, 7, 8

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In

NIPS, pages 1097–1105, 2012. 2, 3

[18] V. Krueger and S. Zhou. Exemplar-based face recognition from video. In *ECCV*, pages 361–375, 2006. 1, 2

[19] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, pages 365–372, 2009. 5

[20] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *CVPR*, pages 3361–3368, 2011. 2, 3

[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2, 3

[22] K. Lee, J. Ho, M. Yang, and D. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *CVPR*, pages 313–320, 2003. 1, 2, 5

[23] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *CVPR*, volume 2, pages 409–415, 2003. 5

[24] L. Liu, Y. Wang, and T. Tan. Online appearance model learning for video-based face recognition. In *CVPR*, pages 17–22, 2007. 1, 2

[25] J. Lu, Y.-P. Tan, and G. Wang. Discriminative multimanifold analysis for face recognition from a single training sample per person. *PAMI*, 35(1):39–51, 2013. 3

[26] J. Lu, G. Wang, W. Deng, and P. Moulin. Simultaneous feature and dictionary learning for image set based face recognition. In *ECCV*, pages 265–280, 2014. 1, 2, 5, 6, 7, 8

[27] J. Lu, G. Wang, and P. Moulin. Image set classification using multiple order statistics features and localized multi-kernel metric learning. In *ICCV*, pages 329–336, 2013. 1, 2, 5, 6, 7, 8

[28] G. Shakhnarovich, J. Fisher, and T. Darrell. Face recognition from long-term observations. In *ECCV*, pages 361–375, 2006. 1, 2

[29] J. Stallkamp, H. Ekenel, and R. Stiefelhagen. Video-based face recognition on real-world data. In *ICCV*, 2007. 1, 2

[30] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *NIPS*, pages 2553–2561, 2013. 2

[31] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1–8, 2014. 2

[32] P. Viola and M. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004. 5

[33] N. Wang and D.-Y. Yeung. Learning a deep compact image representation for visual tracking. In *NIPS*, pages 809–817, 2013. 2

[34] R. Wang and X. Chen. Manifold Discriminant Analysis. In *CVPR*, pages 1–8, 2009. 1, 2, 3, 5, 6, 7, 8

[35] R. Wang, H. Guo, L. Davis, and Q. Dai. Covariance discriminative learning: A natural and efficient approach to image set classification. In *CVPR*, pages 2496–2503, 2012. 1, 2, 5, 6, 7, 8

[36] R. Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image set. In *CVPR*, pages 1–8, 2008. 1, 2, 3, 6, 7, 8

[37] Y. Wu, M. Minoh, M. Mukunoki, and S. Lao. Set based discriminative ranking for recognition. In *ECCV*, pages 497–510, 2012. 1, 2

[38] O. Yamaguchi, K. Fukui, and K. Maeda. Face recognition using temporal image sequence. In *FG*, pages 318–323, 1998. 1, 2, 6, 7, 8

[39] M. Yang, P. Zhu, L. V. Gool, and L. Zhang. Face recognition based on regularized nearest points between image sets. In *FG*, pages 1–7, 2013. 1, 2

[40] P. Zhu, L. Zhang, W. Zuo, and D. Zhang. From point to set: Extend the learning of distance metrics. In *ICCV*, pages 2664–2671, 2013. 1, 2, 6, 7, 8