Correspondence

Learning Prototype Hyperplanes for Face Verification in the Wild

Meina Kan, Dong Xu, *Member, IEEE*, Shiguang Shan, *Member, IEEE*, Wen Li, *Student Member, IEEE*, and Xilin Chen, *Senior Member, IEEE*

***Abstract*— In this paper, we propose a new scheme called Prototype Hyperplane Learning (PHL) for face verification in the wild using only weakly labeled training samples (*i.e.*, we only know whether each pair of samples are from the same class or different classes without knowing the class label of each sample) by leveraging a large number of unlabeled samples in a generic data set. Our scheme represents each sample in the weakly labeled data set as a mid-level feature with each entry as the corresponding decision value from the classification hyperplane (referred to as the prototype hyperplane) of one Support Vector Machine (SVM) model, in which a sparse set of support vectors is selected from the unlabeled generic data set based on the learnt combination coefficients. To learn the optimal prototype hyperplanes for the extraction of mid-level features, we propose a Fisher's Linear Discriminant-like (FLD-like) objective function by maximizing the discriminability on the weakly labeled data set with a constraint enforcing sparsity on the combination coefficients of each SVM model, which is solved by using an alternating optimization method. Then, we use the recent work called Side-Information based Linear Discriminant (SILD) analysis for dimensionality reduction and a cosine similarity measure for final face verification. Comprehensive experiments on two data sets, Labeled Faces in the Wild (LFW) and YouTube Faces, demonstrate the effectiveness of our scheme.**

***Index Terms*— Face verification in the wild, prototype hyperplane learning, mid-level feature representation.**

I. Introduction

In the past two decades, we have witnessed significant progress of face recognition under the controlled conditions and promising results have been reported on data sets including FERET [1], CMU PIE [2], etc., (see [3] for a comprehensive survey). Recently, there is an increasing research interest in face recognition/verification in the wild, in which faces are generally captured in unconstrained conditions (*e.g.*, Flickr photos or YouTube videos). Face verification/recognition in the wild is a more challenging task due to the extremely large within-class appearance variations in terms of pose, illumination, expression, and occlusion.

New methods were recently proposed to improve face verification performance in unconstrained conditions after the release of the Labeled Faces in the Wild (LFW) data set [4]. These methods can be roughly divided into feature based approaches and distance metric based approaches. The feature based approaches aim to develop a better feature representation, among which local feature based methods are more popular. Wolf *et al.* [5] proposed three-patch Local Binary Pattern (LBP) and four-patch LBP features to encode the similarities between neighboring patches around the center pixels in order to capture the information complementary to the LBP feature. In [6], each face was described as multi-region probabilistic histograms of visual words. In [7], Cao *et al.* encoded the micro-structures of each face by using an unsupervised learning approach, while Vu *et al.* [8] developed a discriminative feature descriptor called Patterns of Oriented Edge Magnitudes (POEM) by exploiting the self-similarity of oriented magnitudes. In [9], Pinto *et al.* employed the selected biologically-inspired visual representations for unconstrained face verification. Moreover, several recent works achieved promising results by using similarities among face images as the feature representation. In [10], Kumar *et al.* proposed to use the output of the attributes and simile classifiers as mid-level features for face verification. In [11], Wolf *et al.* used the rank of images that are most similar to a given query image as the descriptor of this query image.

The distance metric based approaches attempt to develop new distance metrics to effectively measure the similarity between two face images. In [5], [12], one-shot similarity was employed to determine whether each sample shares the same class label as its counterpart or belongs to a negative set, which was further extended to two-shot similarity in [11] and multiple one-shot similarity in [13]. In [14], the similarity was calculated from the learnt quantized characteristic difference of local descriptors from a pair of images. A logistic discriminant based distance measure and a nearest neighbor based distance measure were proposed in [15] while a cosine similarity based metric learning method was proposed in [16]. Recently, Yin *et al.* [17] developed a so-called "Associate-Predict" model to measure the similarity between two images by leveraging an extra generic data set with large intra-personal variations. In this model, each face image is associated with visually similar subjects from the generic data set for similarity measurement.

In this work, we propose a new mid-level feature based scheme called Prototype Hyperplane Learning (PHL) for face verification in the wild. Our work is motivated by the recent work in [10], in which the mid-level feature is extracted as the output from a set of pre-learnt SVM models. In contrast to the work in [10] where the SVM models are trained by additionally using a strongly labeled training set (*i.e.*, the class label of each training sample is provided), an additional *unlabeled* generic data set is used in this work to

Manuscript received October 2, 2011; revised February 23, 2013; accepted March 11, 2013. Date of publication April 4, 2013; date of current version June 11, 2013. This work was supported in part by the National Basic Research Program of China's 973 Program under Contract 2009CB320900, and the Natural Science Foundation of China under Contracts 61025010, 61173065, and 61222211. This research was also partially supported by Multi-plAtform Game Innovation Centre (MAGIC) in Nanyang Technological University. MAGIC is funded by the Interactive Digital Media Programme Office (IDMPO) hosted by the Media Development Authority of Singapore. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. A. N. Rajagopalan.

M. Kan, S. Shan, and X. Chen are with the Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology (ICT), CAS, Beijing 100190, China (e-mail: meina.kan@vipl.ict.ac.cn; shiguang.shan@vipl.ict.ac.cn; xilin.chen@vipl.ict.ac.cn).

D. Xu and W. Li are with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: dongxu@ntu.edu.sg; wli1@e.ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2013.2256918

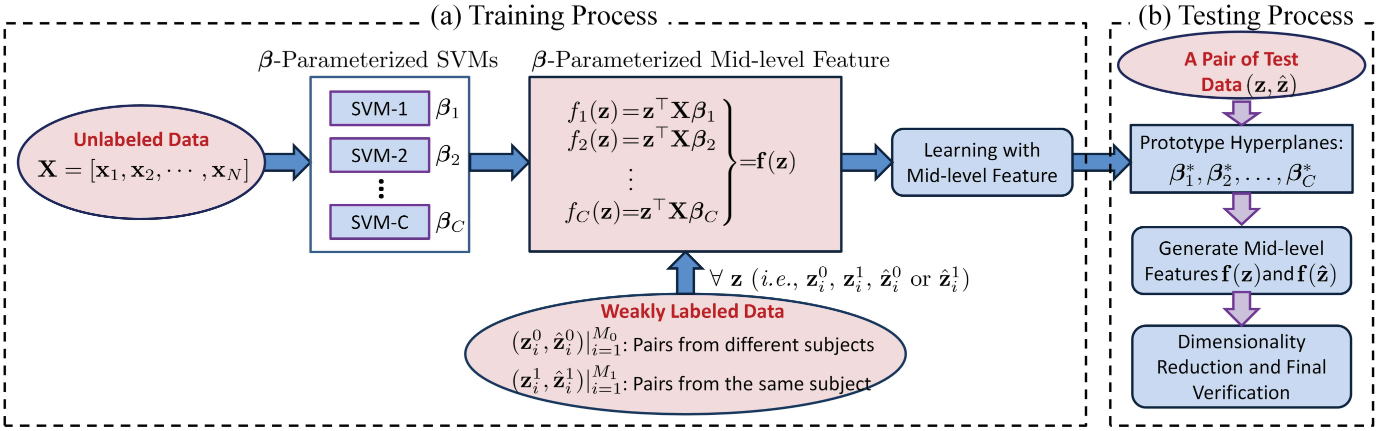

(a) Training Process
(b) Testing Process
β-Parameterized SVMs
β-Parameterized Mid-level Feature
Unlabeled Data
$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$
SVM-1 β_1
SVM-2 β_2
SVM-C β_C
$f_1(\mathbf{z}) = \mathbf{z}^\top \mathbf{X}\beta_1$
$f_2(\mathbf{z}) = \mathbf{z}^\top \mathbf{X}\beta_2$
$f_C(\mathbf{z}) = \mathbf{z}^\top \mathbf{X}\beta_C$
$= \mathbf{f}(\mathbf{z})$
Learning with Mid-level Feature
$\forall\ \mathbf{z}\ (i.e.,\ \mathbf{z}_i^0,\ \mathbf{z}_i^1,\ \hat{\mathbf{z}}_i^0$ or $\hat{\mathbf{z}}_i^1)$
Weakly Labeled Data
$(\mathbf{z}_i^0, \hat{\mathbf{z}}_i^0)|_{i=1}^{M_0}$: Pairs from different subjects
$(\mathbf{z}_i^1, \hat{\mathbf{z}}_i^1)|_{i=1}^{M_1}$: Pairs from the same subject
A Pair of Test Data $(\mathbf{z}, \hat{\mathbf{z}})$
Prototype Hyperplanes: $\beta_1^*, \beta_2^*, \ldots, \beta_C^*$
Generate Mid-level Features $\mathbf{f}(\mathbf{z})$ and $\mathbf{f}(\hat{\mathbf{z}})$
Dimensionality Reduction and Final Verification

Fig. 1. Illustration of our Prototype Hyperplane Learning scheme: (a) Training process; (b) Testing process.

construct the SVM models (*i.e.*, prototype hyperplanes). The mid-level feature representation can be obtained by using the prototype hyperplanes. Then, we formulate a new Fisher's Linear Discriminant-like (FLD-like) [23] objective function by using a weakly labeled data set (*i.e.*, we only know whether a pair of samples are from the same subject or different subjects without knowing the exact class label of each sample). We learn the optimal prototype hyperplanes by maximizing the FLD-like objective on the weakly labeled data set with a sparsity constraint in each SVM model, which selects only a sparse set of support vectors from the generic data set. Inspired by [18], we develop an alternating optimization algorithm to solve our objective function and the resultant non-zero combination coefficients automatically decide each prototype hyperplane. Finally, the recent work SILD [19] is used to reduce the feature dimension and the cosine similarity is employed for the final face verification. We conduct comprehensive experiments using two real world face data sets, Labeled Faces in the Wild (LFW) and YouTube Faces, and the results demonstrate the effectiveness of our scheme for face verification in unconstrained conditions.

II. Prototype Hyperplane Learning

In this section, we present the details of our Prototype Hyperplane Learning scheme including problem formulation and optimization. In this work, we use boldface lowercase and uppercase letters to denote a vector (*e.g.* $\mathbf{a}$) and a matrix (*e.g.* $\mathbf{A}$), respectively. We also define $\mathbf{I}$ and $\mathbf{0}$ as an identity matrix and a column vector with all entries being 0, respectively.

A. Problem Formulation

Let us denote an unlabeled generic data set $\mathcal{X} = \{\mathbf{x}_i|_{i=1}^N\}$ with its data matrix represented as $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$, where D is the feature dimension and N is the total number of samples in this data set. We also define a weakly labeled data set consisting of M_1 pairs of samples $\{(\mathbf{z}_i^1, \hat{\mathbf{z}}_i^1)|_{i=1}^{M_1}\}$ from the same subject and M_0 pairs of samples $\{(\mathbf{z}_i^0, \hat{\mathbf{z}}_i^0)|_{i=1}^{M_0}\}$ from different subjects, where the class label of each sample is unknown and the feature dimension of each sample is also D. In this work, we aim to learn a few classification hyperplanes of binary SVM models by using the weakly labeled data set, in which a sparse set of support vectors are automatically selected from the unlabeled generic data set. Each sample in the weakly labeled data set is represented as a mid-level feature with each entry as the corresponding decision value from one learnt SVM model. Then, we propose an FLD-like objective function to learn the optimal prototype hyperplanes by maximizing the discriminability on the weakly labeled data set with a sparsity constraint that selects only a sparse set of support vectors from the generic data set in each SVM model. The process of learning the prototype hyperplanes is illustrated in Fig. 1(a).

1) Mid-Level Feature Representation from Prototype Hyperplanes: In this work, each prototype hyperplane is modeled by using a linear SVM with the support vectors automatically chosen from the large unlabeled generic data set $\mathcal{X}$. Note that our linear SVM model can be readily extended to a non-linear one by using the kernel trick. For each linear SVM model, the weight vector $\mathbf{w}$ for the feature can be formulated as follows by using the Representer Theorem:

$$\mathbf{w} = \sum\nolimits_{j=1}^{N} \alpha_j y_j \mathbf{x}_j = \sum\nolimits_{j=1}^{N} \beta_j \mathbf{x}_j = \mathbf{X}\boldsymbol{\beta}, \tag{1}$$

where α_j and y_j are the dual variable and the inferred class label of the unlabeled data $\mathbf{x}_j$ respectively, the combination coefficient $\beta_j = \alpha_j y_j$ $(j = 1, 2, \ldots, N)$ merges the dual variable and inferred class label of each unlabeled sample, and a combination coefficient vector is defined as $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_N]^T \in \mathbb{R}^N$.

In our work, $\mathbf{x}_j$ is an augmented low-level feature (*e.g.*, Gabor or LBP feature) with the last entry as one in order to avoid introducing the bias term in the SVM model. The optimal classification hyperplane of each SVM model is decided by the learnt combination coefficients $\beta_j (j = 1, \ldots, N)$. Specifically, if β_j is non-zero, the unlabeled sample $\mathbf{x}_j$ in the generic data set is chosen as a support vector of the SVM model. While the support vectors are chosen from the unlabeled generic data set $\mathcal{X}$, the label of each support vector can also be inferred after the learning process. If β_j is positive (*resp.* negative), we have $y_j = 1$ (*resp.* $y_j = -1$) and $\mathbf{x}_j$ is actually used as a positive sample (*resp.* negative sample) in the SVM model. Moreover, each classification hyperplane of the SVM model is expected to lie in the margin between two classes, which means we only select a sparse set of support vectors. In order to select only a sparse set of samples from the generic data set as support vectors, we also enforce $\boldsymbol{\beta}$ to be a sparse vector, namely $\|\boldsymbol{\beta}\|_1 \leq t$, where t is a parameter for controlling the sparsity of $\boldsymbol{\beta}$.

Given any sample $\mathbf{z}$ in the weakly labeled data set, its decision value from the SVM model is:

$$f(\mathbf{z}) = \mathbf{w}^T \mathbf{z} = \mathbf{z}^T \mathbf{w} = \mathbf{z}^T \mathbf{X}\boldsymbol{\beta} \tag{2}$$

which measures the likelihood of the sample $\mathbf{z}$ according to the SVM model. Suppose we have C linear SVM models, then we seek for a combination coefficient vector $\boldsymbol{\beta}_i (i = 1, \ldots, C)$ for each SVM model. Let us define a combination coefficient matrix $\mathbf{B} = [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_C] \in \mathbb{R}^{N \times C}$, then the mid-level feature of a sample $\mathbf{z}$ can be represented as:

$$\begin{aligned}\mathbf{f}(\mathbf{z}) &= [\mathbf{z}^T\mathbf{X}\boldsymbol{\beta}_1, \mathbf{z}^T\mathbf{X}\boldsymbol{\beta}_2, \ldots, \mathbf{z}^T\mathbf{X}\boldsymbol{\beta}_C]^T \\ &= (\mathbf{z}^T\mathbf{X}\mathbf{B})^T = \mathbf{B}^T\mathbf{X}^T\mathbf{z}. \end{aligned} \tag{3}$$

Since the mid-level feature representation depends on the parameter $\boldsymbol{\beta}_i|_{i=1}^{C}$, we refer to the feature $\mathbf{f}(\mathbf{z})$ as $\boldsymbol{\beta}$-parameterized mid-level feature.

2) Learning with the Mid-Level Feature: Using the new mid-level feature representation $\mathbf{f}(\mathbf{z})$ of each training sample $\mathbf{z}$ in the weakly labeled data set, we propose an FLD-like criterion to learn the optimal combination coefficient matrix $\mathbf{B}$. Note that our method can also work when the class label of each sample is provided. Specifically, we propose the following objective function to learn the optimal $\mathbf{B}$ by minimizing the intra-class scatter and at the same time maximizing the inter-class scatter on the weakly labeled data:

$$\begin{aligned}\mathbf{B}^* &= \arg\max_{\mathbf{B}} G(\mathbf{B}) = \arg\max_{\mathbf{B}} \frac{\sum_{i=1}^{M_0} \|\mathbf{f}(\mathbf{z}_i^0) - \mathbf{f}(\hat{\mathbf{z}}_i^0)\|^2}{\sum_{i=1}^{M_1} \|\mathbf{f}(\mathbf{z}_i^1) - \mathbf{f}(\hat{\mathbf{z}}_i^1)\|^2}, \\ & s.t.\ \|\boldsymbol{\beta}_i\|_1 \le t, \quad i = 1, \ldots, C. \end{aligned} \tag{4}$$

In Eq. (4), the numerator measures the inter-class distance for all pairs of training samples $(\mathbf{z}_i^0, \hat{\mathbf{z}}_i^0)|_{i=1}^{M_0}$ from different subjects while the denominator measures the intra-class distance for all pairs of training samples $(\mathbf{z}_i^1, \hat{\mathbf{z}}_i^1)|_{i=1}^{M_1}$ from the same subject. Again, we enforce the sparsity constraint on $\boldsymbol{\beta}_i$ in order to only select a sparse set of support vectors from the unlabeled data set. In this work, we use the same parameter t for different $\boldsymbol{\beta}_i$ in order to facilitate model selection. By using Eq. (3) and the property $\|\mathbf{A}\|^2 = \mathrm{Tr}(\mathbf{A}\mathbf{A}^T)$, we rewrite $G(\mathbf{B})$ in Eq. (4) as follows:

$$\begin{aligned} G(\mathbf{B}) &= \frac{\sum_{i=1}^{M_0} \|\mathbf{B}^T\mathbf{X}^T\mathbf{z}_i^0 - \mathbf{B}^T\mathbf{X}^T\hat{\mathbf{z}}_i^0\|^2}{\sum_{i=1}^{M_1} \|\mathbf{B}^T\mathbf{X}^T\mathbf{z}_i^1 - \mathbf{B}^T\mathbf{X}^T\hat{\mathbf{z}}_i^1\|^2} \\ &= \frac{\sum_{i=1}^{M_0} \mathrm{Tr}(\mathbf{B}^T\mathbf{X}^T(\mathbf{z}_i^0 - \hat{\mathbf{z}}_i^0)(\mathbf{z}_i^0 - \hat{\mathbf{z}}_i^0)^T\mathbf{X}\mathbf{B})}{\sum_{i=1}^{M_1} \mathrm{Tr}(\mathbf{B}^T\mathbf{X}^T(\mathbf{z}_i^1 - \hat{\mathbf{z}}_i^1)(\mathbf{z}_i^1 - \hat{\mathbf{z}}_i^1)^T\mathbf{X}\mathbf{B})} \\ &= \frac{\mathrm{Tr}(\mathbf{B}^T\mathbf{S}_b\mathbf{B})}{\mathrm{Tr}(\mathbf{B}^T\mathbf{S}_w\mathbf{B})}, \end{aligned} \tag{5}$$

where $\mathbf{S}_b$ and $\mathbf{S}_w$ are defined as

$$\begin{aligned}\mathbf{S}_b &= \sum\nolimits_{i=1}^{M_0} \mathbf{X}^T(\mathbf{z}_i^0 - \hat{\mathbf{z}}_i^0)(\mathbf{z}_i^0 - \hat{\mathbf{z}}_i^0)^T\mathbf{X}, \\ \mathbf{S}_w &= \sum\nolimits_{i=1}^{M_1} \mathbf{X}^T(\mathbf{z}_i^1 - \hat{\mathbf{z}}_i^1)(\mathbf{z}_i^1 - \hat{\mathbf{z}}_i^1)^T\mathbf{X}. \end{aligned} \tag{6}$$

According to [20], [21], the objective function in Eq. (5) is in the trace ratio form, for which the closed form solution does not exist. We therefore reformulate the trace ratio problem in Eq. (5) into a more tractable ratio trace form and arrive at:

$$\begin{aligned}\mathbf{B}^* = \arg\max_{\mathbf{B}} Tr\left((\mathbf{B}^T\mathbf{S}_w\mathbf{B})^{-1}(\mathbf{B}^T\mathbf{S}_b\mathbf{B})\right), \quad & s.t.\ \|\boldsymbol{\beta}_i\|_1 \le t, \\ & i = 1, \ldots, C. \end{aligned} \tag{7}$$

Note that generalized eigenvalue decomposition method can be directly used to solve the ratio trace problem in Eq. (7), if there is no constraint for $\boldsymbol{\beta}_i$ $(i = 1, 2, \ldots, C)$. Moreover, due to the sparsity constraint for $\boldsymbol{\beta}_i (i = 1, \ldots, C)$ in Eq. (7), the existing methods in [20], [21] for the trace ratio problem cannot be employed to solve our problem, either. Therefore, in this work we use an alternating optimization method in [18] to solve for the optimal $\mathbf{B}$.

B. Optimization

We first reformulate the objective function in Eq. (7) from the ratio trace problem into a regression problem, which can be solved by using the alternating optimization method.

1) Reformulate the Ratio Trace Problem in Eq. (7) as a Regression Problem: Given the M_0 pairs of samples $\{(\mathbf{z}_i^0, \hat{\mathbf{z}}_i^0)|_{i=1}^{M_0}\}$ from different subjects in the weakly labeled data set, let us define two data matrices as $\mathbf{D} = \left[(\mathbf{z}_1^0 - \hat{\mathbf{z}}_1^0), (\mathbf{z}_2^0 - \hat{\mathbf{z}}_2^0), \ldots, (\mathbf{z}_{M_0}^0 - \hat{\mathbf{z}}_{M_0}^0)\right] \in \mathbb{R}^{D \times M_0}$ and $\mathbf{H}_b = \mathbf{D}^T\mathbf{X} \in \mathbb{R}^{M_0 \times N}$. We also conduct Singular Value Decomposition (SVD) of $\mathbf{S}_w$ in Eq. (6), *i.e.*, $\mathbf{S}_w = \mathbf{R}_w^T\mathbf{R}_w$, to define another matrix $\mathbf{R}_w \in \mathbb{R}^{N \times N}$. Following [18], we reformulate the ratio trace problem as a regression problem by introducing an intermediate variable $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_C] \in \mathbb{R}^{N \times C}$ (please refer to [18] for more details on the reformulation):

$$\begin{aligned}\left[\mathbf{A}^*, \mathbf{B}^*\right] &= \arg\min_{\mathbf{A},\mathbf{B}} \sum_{i=1}^{C} \|\mathbf{H}_b\mathbf{R}_w^{-1}\mathbf{a}_i - \mathbf{H}_b\boldsymbol{\beta}_i\|^2 + \sum_{i=1}^{C} \lambda\boldsymbol{\beta}_i^T\mathbf{S}_w\boldsymbol{\beta}_i, \\ & s.t.\ \mathbf{A}^T\mathbf{A} = \mathbf{I}_{C \times C}, \|\boldsymbol{\beta}_i\|_1 \le t, \quad i = 1, \ldots, C. \end{aligned} \tag{8}$$

2) Optimize the Regression Problem in Eq. (8): As suggested in [18], we employ an alternating optimization method to iteratively optimize $\mathbf{A}$ and $\mathbf{B}$. Given $\mathbf{A}$, we solve the following problem to obtain $\mathbf{B}$:

$$\begin{aligned}\mathbf{B}^* &= \arg\min_{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_C} \sum_{i=1}^{C} \left(\|\mathbf{H}_b\mathbf{R}_w^{-1}\mathbf{a}_i - \mathbf{H}_b\boldsymbol{\beta}_i\|^2 + \lambda\boldsymbol{\beta}_i^T\mathbf{S}_w\boldsymbol{\beta}_i\right), \\ & s.t.\ \|\boldsymbol{\beta}_i\|_1 \le t, \quad i = 1, \ldots, C. \end{aligned} \tag{9}$$

Observing that $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_C$ are independent in Eq. (9), we separately solve for each $\boldsymbol{\beta}_i$ by optimizing the following problem:

$$\begin{aligned}\boldsymbol{\beta}_i^* &= \arg\min_{\boldsymbol{\beta}_i} \|\mathbf{s}_i - \mathbf{H}_b\boldsymbol{\beta}_i\|^2 + \lambda\boldsymbol{\beta}_i^T\mathbf{S}_w\boldsymbol{\beta}_i \\ &= \arg\min_{\boldsymbol{\beta}_i} \|\tilde{\mathbf{s}}_i - \tilde{\mathbf{W}}\boldsymbol{\beta}_i\|^2, \quad s.t.\ \|\boldsymbol{\beta}_i\|_1 \le t, \end{aligned} \tag{10}$$

with

$$\mathbf{s}_i = \mathbf{H}_b\mathbf{R}_w^{-1}\mathbf{a}_i, \tilde{\mathbf{s}}_i = \left[\mathbf{s}_i^T, \mathbf{0}_N^T\right]^T \quad \text{and} \quad \tilde{\mathbf{W}} = \left[\mathbf{H}_b^T, \sqrt{\lambda}\mathbf{R}_w^T\right]^T.$$

The Least Angle Regression solver [22] is employed to solve for the optimal $\boldsymbol{\beta}_i$ in this work.

Given $\mathbf{B}$, we can ignore the constraint on $\boldsymbol{\beta}_i$ and directly compute $\mathbf{A}$ by solving the following problem:

$$\begin{aligned}\mathbf{A}^* &= \arg\min_{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_C} \sum\nolimits_{i=1}^{C} \|\mathbf{H}_b\mathbf{R}_w^{-1}\mathbf{a}_i - \mathbf{H}_b\boldsymbol{\beta}_i\|^2, \\ &= \arg\min_{\mathbf{A}} \|\mathbf{H}_b\mathbf{R}_w^{-1}\mathbf{A} - \mathbf{H}_b\mathbf{B}\|^2, \quad s.t.\ \mathbf{A}^T\mathbf{A} = \mathbf{I}_{C \times C} \end{aligned} \tag{11}$$

The optimal $\mathbf{A}$ can be obtained by using SVD, namely

$$\mathbf{R}_w^{-T}\left(\mathbf{H}_b^T\mathbf{H}_b\right)\mathbf{B} = \mathbf{U}\Sigma\mathbf{V}^T, \quad \text{and} \quad \mathbf{A}^* = \tilde{\mathbf{U}}\mathbf{V}^T \tag{12}$$

where $\tilde{\mathbf{U}} = \left[\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_C\right]$ contains the first C leading eigenvectors of the matrix $\mathbf{U} = \left[\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N\right]$. In this work, we iteratively solve Eq. (9) and (11) until the absolute difference of $\mathbf{B}$ from two successive iterations is smaller than a pre-defined threshold. The detailed algorithm is listed in Table I.

C. Dimensionality Reduction using SILD and Final Verification

With the learnt prototype hyperplanes, each sample can be represented as its mid-level decision values feature using Eq. (3). To further reduce the feature dimension and improve the performance, we employ our recent work SILD [19] for dimensionality reduction,

TABLE I

THE PROTOTYPE HYPERPLANE LEARNING (PHL) ALGORITHM

Inputs: The unlabeled generic data set $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$, a weakly labeled data set consisting of M_1 pairs of samples $\{(\mathbf{z}_i^1, \hat{\mathbf{z}}_i^1)|_{i=1}^{M_1}\}$ from the same subject, and M_0 pairs of samples $\{(\mathbf{z}_i^0, \hat{\mathbf{z}}_i^0)|_{i=1}^{M_0}\}$ from different subjects with the corresponding data matrix defined as $\mathbf{D} = \left[(\mathbf{z}_1^0 - \hat{\mathbf{z}}_1^0), (\mathbf{z}_2^0 - \hat{\mathbf{z}}_2^0), \ldots, (\mathbf{z}_{M_0}^0 - \hat{\mathbf{z}}_{M_0}^0)\right] \in \mathbb{R}^{D \times M_0}$

Result: The optimal combination coefficient vectors $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_C \in \mathbb{R}^{N \times 1}$ that determine the classification hyperplanes of SVM models

Initialization: Initialize $\mathbf{A} \in \mathbb{R}^{N \times C}$ and $\mathbf{B} \in \mathbb{R}^{N \times C}$ with all entries as 1, and calculate $\mathbf{S}_b$ and $\mathbf{S}_w$ using Eq. (6).
Calculate $\mathbf{H}_b = \mathbf{D}^T \mathbf{X} \in \mathbb{R}^{M_0 \times N}$ and $\mathbf{R}_w \in \mathbb{R}^{N \times N}$ by conducting SVD of $\mathbf{S}_w$, *i.e.*, $\mathbf{S}_w = \mathbf{R}_w^T \mathbf{R}_w$

Repeat

Given $\mathbf{A}$, solve C independent Lasso problems in Eq. (10) using the Least Angle Regression solver [22]:
$\boldsymbol{\beta}_i^* = \arg\min_{\boldsymbol{\beta}_i} \|\tilde{\mathbf{s}}_i - \tilde{\mathbf{W}}\boldsymbol{\beta}_i\|^2, \ s.t. \ \|\boldsymbol{\beta}_i\|_1 \leq t, i = 1, \ldots, C.$
Given $\mathbf{B}$, conduct SVD, *i.e.*, $\mathbf{R}_w^{-T}\left(\mathbf{H}_b^T \mathbf{H}_b\right)\mathbf{B} = \mathbf{U}\Sigma\mathbf{V}^T$, and solve for $\mathbf{A}^*$ by using $\mathbf{A}^* = \tilde{\mathbf{U}}\mathbf{V}^T$, where $\tilde{\mathbf{U}} = \left[\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_C\right]$ contains the first C leading eigenvectors of the matrix $\mathbf{U} = \left[\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N\right]$

Until: *The change of* $\mathbf{B}$ *between two successive iterations is smaller than* ε ($\varepsilon = 0.001$ *in this work).*

which can learn a discriminative projection matrix by using only weakly labeled training data. SILD is proven to be equivalent to Fisher's Linear Discriminant Analysis [23] when the class label information of each sample is available [19]. Specifically, in the training process of SILD [19], the within-class scatter matrix is defined by only using the pairs of samples from the same subject and the between-class scatter matrix is defined by only using the pairs of samples from different subjects. After that, generalized eigenvalue decomposition is employed to determine the projection matrix for dimensionality reduction. In the testing process (see Fig. 1(b)), for each pair of test data $\mathbf{z}$ and $\hat{\mathbf{z}}$, we respectively generate the mid-level feature representations $\mathbf{f}(\mathbf{z})$ and $\mathbf{f}(\hat{\mathbf{z}})$ by using the learnt prototype hyperplanes, and then map the mid-level features $\mathbf{f}(\mathbf{z})$ and $\mathbf{f}(\hat{\mathbf{z}})$ into a low dimensional space by using the projection matrix learnt in the training process of SILD. Finally, the cosine function is used to calculate the similarity for a pair of test samples before conducting face verification. The whole process is illustrated in Fig. 1.

D. Discussion of Existing Work

While our method and the recent work in [10] both employ the decision values from a large number of SVM models as the mid-level feature representation for face verification, our work is intrinsically different from [10]. The SVM models in [10] are from attributes classifiers which require substantial manual labeling effort and simile classifiers trained by additionally using strongly labeled face images. In contrast, in our work the classification hyperplanes of the SVM models are decided according to an FLD-like objective function using the weakly labeled data set in which the support vectors are automatically chosen from a large unlabeled data set.

Our work is also different from the SVM based semi-supervised learning methods like Transductive SVM (TSVM) [24] based on the cluster assumption and Laplacian SVM (LapSVM) [25] based on the manifold assumption. In most semi-supervised learning methods (see [26] for a recent survey) including TSVM and LapSVM, both strongly labeled training samples (*i.e.*, the class label of each training sample is provided) and unlabeled training samples are required. In contrast, in our work we only use weakly labeled training samples and an unlabeled generic data set.

III. Experiments

In this section, we compare our proposed Prototype Hyperplane Learning (PHL) scheme with the state-of-the-art methods on two data sets, Labeled Faces in the Wild (LFW) [4] and YouTube Faces [27], which are both collected in unconstrained conditions.

A. Data Set Descriptions and Experimental Settings

The LFW database [4] is a large data set consisting of 13,233 images from 5,749 individuals. The standard evaluation protocol has two views, in which view 1 is employed for model selection, and view 2 is used for performance evaluation. In our experiments, the center area of each face image provided in [11] is cropped to an image of 80 × 150 pixels by removing the background as suggested in [16]. The YouTube Faces Database [27] is a large unconstrained video data set, which contains 3,425 videos from 1,595 subjects. On average, there are 2.15 videos for each subject and the length of each video clip is about 181 frames at 24 fps.

On both data sets, we use the so-called image-restricted training mode, *i.e.*, we only know whether a pair of samples are from the same subject or different subjects without knowing the class label of each sample. To construct the unlabeled generic data set $\mathcal{X}$, 3,000 unlabeled samples are randomly selected from view 1 on the LFW data set, and from the training set on the YouTube Faces data set. It is worth mentioning there are no overlapping images between the unlabeled generic data set and test set because we intend not to select the overlapping images when constructing the generic data set. In all experiments, the number of prototype hyperplanes C is set as 400. On the LFW data set, the optimal parameter t in Eq. (7) is determined by using cross validation based on the data from view 1, while this parameter is empirically set as 0.5 on the YouTube Faces data set because there is no additional data set for model selection. We also take the YouTube Faces data set as an example to investigate the performance variations of our PHL with respect to the parameters C and t (see Section III-C).

We report the mean accuracy with the standard error (SE)/standard deviation (std) and the ROC curve from ten-fold cross validation according to the standard protocol [4], [27]. Given the learnt threshold determined from the training data, the accuracy at each round of the experiment is defined as the number of correctly classified pairs of samples divided by the total number of test sample pairs. The standard error is defined as $\hat{\sigma}/\sqrt{10}$, where $\hat{\sigma}$ is the standard deviation.

B. Comparison with the State-of-the-Art Results

We compare our PHL with the state-of-the-art methods on the LFW and YouTube Faces data sets.

1) Results on the LFW Database: On the LFW data set, we use eight types of features including Intensity, LBP, Gabor feature and Block Gabor feature as well as the square root of these features as suggested in [11], [16], [19]. The intensity feature is

TABLE II

PERFORMANCES (MEAN ACCURACY±STANDARD ERROR) OF OUR PHL+SILD AND LOW-LEVEL FEATURE+SILD [19] USING DIFFERENT TYPES OF LOW-LEVEL FEATURES ON THE LFW DATA SET

Feature Name	Feature Type	Low-level Feature +SILD [19]	PHL+SILD (this work)
Intensity	Original	0.8020 ± 0.0067	0.8097 ± 0.0072
	Square root	0.8010 ± 0.0056	0.7925 ± 0.0045
LBP	Original	0.8412 ± 0.0034	0.8442 ± 0.0062
	Square root	0.8485 ± 0.0035	0.8542 ± 0.0064
Gabor	Original	0.7902 ± 0.0059	0.8130 ± 0.0065
	Square root	0.8102 ± 0.0064	0.8355 ± 0.0056
Block Gabor	Original	0.8233 ± 0.0052	0.8343 ± 0.0067
	Square root	0.8452 ± 0.0044	0.8510 ± 0.0052
Combined Results		**0.8768 ± 0.0050**	**0.8867 ± 0.0070**

TABLE III

PERFORMANCES (MEAN ACCURACY±STANDARD ERROR (SE)) OF OUR PHL+SILD AND OTHER STATE-OF-THE-ART ALGORITHMS ON THE LFW DATA SET

Type	Methods	Mean Acc. ± SE
Without additional data	Multiregion Histograms [6]	0.7295 ± 0.0055
	Multiple LE + comp [7]	0.8445 ± 0.0046
	Low-level Feature+SILD [19]	0.8768 ± 0.0050
	CSML + SVM [16]	0.8800 ± 0.0037
	High-Throughput Brain-Inspired Features [9]	0.8813 ± 0.0058
With labeled additional data	Attribute and Simile classifiers [10]	0.8529 ± 0.0123
	Associate-Predict [17]	0.9057 ± 0.0056
With unlabeled additional data	Combined b/g samples based methods [11]	0.8683 ± 0.0034
	PHL+SILD (this work)	**0.8867 ± 0.0070**

TABLE IV

PERFORMANCES (MEAN ACCURACY± STANDARD DEVIATION (STD), AUC AND EER) OF OUR PHL+SILD, LOW-LEVEL FEATURE+SILD [19] AND MBGS [27] USING LBP, CSLBP AND FPLBP FEATURES ON THE YOUTUBE FACES DATA SET

Feature	Methods	Mean Acc. ± std	AUC	EER
LBP	MBGS [27]	0.764 ± 0.018	0.826	0.253
	Low-level Feature+SILD [19]	0.773 ± 0.019	0.840	0.236
	PHL+SILD (this work)	**0.802 ± 0.013**	**0.872**	**0.203**
CSLBP	MBGS [27]	0.724 ± 0.020	0.789	0.287
	Low-level Feature+SILD [19]	0.736 ± 0.015	0.804	0.286
	PHL+SILD (this work)	**0.752 ± 0.010**	**0.823**	**0.248**
FPLBP	MBGS [27]	0.726 ± 0.020	0.801	0.277
	Low-level Feature+SILD [19]	0.729 ± 0.024	0.796	0.283
	PHL+SILD (this work)	**0.759 ± 0.015**	**0.825**	**0.244**

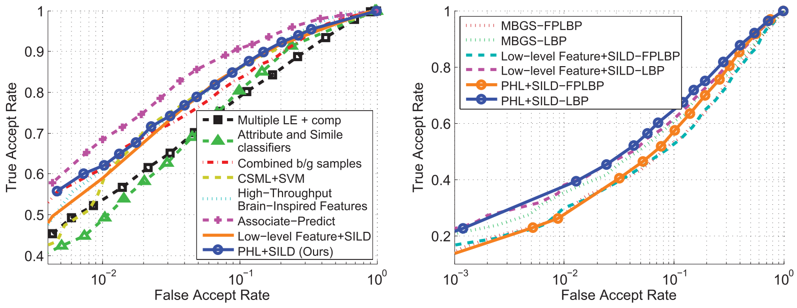
directly extracted by vectorizing each gray-scale image to a 12,000 dimensional feature vector. For the LBP feature, a histogram of 59 bins is first extracted from each non-overlapping block of 10x10 pixels, and then all histograms are concatenated into a single 7,080 dimensional feature vector. We use 40 Gabor kernel functions from 5 scales and 8 orientations to extract the Gabor feature [28] and the Gabor filtered images are further downsampled by using a 10 × 10 scaling factor [16] in order to reduce the feature dimension. However, such a significant downsampling operation may degrade the face verification performance. Following [29], we additionally use the block Gabor feature by dividing each Gabor filtered image into 6 non-overlapping blocks before downsampling and the Gabor filtered sub-images at each block are only downsampled by using a 2×2 scaling factor. We then treat the Gabor features in each block separately rather than concatenating them into one lengthy feature vector. In other words, for the Gabor features in each block, we apply our PHL for extracting mid-level features, followed by SILD for dimensionality reduction. After that, for each pair of face images, the six similarities after using the cosine function for the Gabor features from all the six blocks are averaged to output one similarity score only. To fuse eight types of features, each pair of images is represented as an 8-dimensional similarity feature and a linear SVM is further employed to calculate the final similarity for each pair of images.

As mentioned in Section II-C, we employ our recent work SILD [19] for dimensionality reduction because it is generally beneficial to conduct dimensionality reduction before the final verification. We therefore refer to our scheme discussed in this work and the method in [19] as "PHL+SILD" and "Low-level feature+SILD," respectively. It is worth mentioning the difference is that we use the mid-level features (*i.e.*, the decision values from the learnt SVM models) in our PHL+SILD rather than the original low-level features in Low-level Feature+SILD [19]. The results are shown in Table II. Except for the square root of Intensity feature, our PHL+SILD outperforms Low-level feature+SILD [19] for other types of features and the performance improves up to 2.53% when using the square root of Gabor feature, which demonstrates the effectiveness of using our proposed PHL scheme to learn the optimal classification hyperplane for extracting the mid-level features. When compared with the single feature based method "Single LE" [7] whose performance is 81.22%, the result of our PHL+SILD using the square root of LBP feature is 85.42%, which is much better. Moreover, our PHL+SILD using all eight types of features can achieve the best result of 88.67%.

We also compare our method with state-of-the-art methods including "Multi-Region Histogram" [6], "Combined b/g samples based method" [11], "Attribute and Simile Classifiers" [10], "Multi-LE+comp" [7], "CSML+SVM" [16], "High-Throughput Brain-Inspired Feature" [9] and "Associate-Predict" [17] (for all the results, please refer to http://vis-www.cs.umass.edu/lfw/results.html). All the results are shown in Table III and we also report the ROC curves in Fig. 2(a). Our PHL+SILD is better than all the existing methods without using additional data [6], [7], [9], [16], [19]. Our PHL+SILD also outperforms the work in [11] which uses unlabeled additional data and the mid-level feature based method [10] which additionally uses strongly labeled training data to learn SVM classifiers. Our work is only worse than the recent work "Associate Predict" [17], in which a strongly labeled additional data set with extensive intra-personal variations is required. In contrast, only an additional unlabeled data set is needed in our PHL+SILD.

2) Results on the YouTube Faces Database: On this data set, we directly use the three types of features (*i.e.*, LBP, CSLBP, and FPLBP) provided in [27]. Considering that all the faces are aligned by fixing the detected facial key points [27], the features extracted from all the frames within one video clip are averaged to output a mean feature vector for further processing in our PHL+SILD and low-level feature+SILD methods.

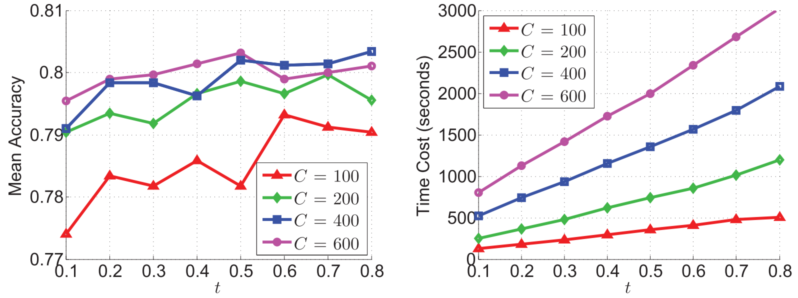
We compare our PHL+SILD with the state-of-the-art method "MBGS" [27] and our recent work "Low-level Feature+SILD" [19] in Table IV in terms of Mean Accuracy, Area under Curve (AUC) and Equal Error Rate (EER). Compared with "Low-level feature+SILD" [19], our work PHL+SILD is still better when using the three types of features, and the performance improves up to 3.0% in terms of mean accuracy, which again demonstrates that it is beneficial to use our PHL scheme to learn the classification



(a) ROC curves on LFW

(b) ROC curves on YouTube Faces

Fig. 2. ROC curves of different approaches on the LFW and YouTube Faces data sets (best viewed in color).



(a) The mean accuracy

(b) The training time (seconds)

Fig. 3. The mean accuracy and training time of our PHL+SILD when using different numbers of prototype hyperplanes (*i.e.*, C) and sparsity parameters t on the YouTube Faces data set with the LBP feature (best viewed in color).

hyperplanes for extracting mid-level features. Using the LBP feature, the improvements of our PHL+SILD over MBGS are 3.8%, 4.6%, 5.0% in terms of ACC, AUC and EER, respectively. Using the CSLBP and FPLBP features, our method PHL+SILD is also better than MBGS. Fig. 2(b) plots the ROC curves of the work in [27] and our PHL+SILD method using the LBP and FPLBP features. From Fig. 2(b), it can be observed that our method PHL+SILD generally outperforms MBGS.

C. Discussion on the Parameters

In Fig. 3, we take the YouTube Faces data set using the LBP feature as an example to study the performance variations of our PHL+SILD with respect to the two parameters C and t, in which we set $C = 100, 200, 400, 600$. Considering that our initial experiments show that the resultant $\boldsymbol{\beta}_i$ will become non-sparse and the computational cost will significantly increase when setting t larger than 0.8, we set $t = 0.1, 0.2, \ldots, 0.8$ in this work. The experiments are conducted on a desktop (3.10 GHz CPU with 8 GB RAM).

When setting C to a larger number, the mean accuracies of our PHL+SILD generally become better and at the same time the training time also increases (see Fig. 3). We also have similar observations on the YouTube Faces data set using other features and on the LFW data set. For the tradeoff between efficiency and effectiveness, we empirically set $C = 400$ on both data sets when using all types of features. Moreover, the results of our work become relatively stable when setting the parameter t between 0.2 and 0.8. Considering that there is no pre-defined additional data set for model selection, we therefore empirically fix the parameter t as 0.5 on the YouTube Faces data set. Following existing work [4], [10], [11], [17], the optimal parameter t on the LFW data set is decided by using cross validation with the data from view 1.

IV. Conclusion

In this work, we have proposed a new scheme called Prototype Hyperplane Learning (PHL) to seek a mid-level feature representation for face verification in the wild by learning a set of prototype hyperplanes of SVM models, in which the support vectors of each SVM model are chosen from a large unlabeled generic data set. We propose an FLD-like objective function to optimize the optimal prototype hyperplanes by maximizing the discriminability on the weakly labeled data set with a sparsity constraint that selects only a sparse set of samples from the generic data set as support vectors. The decision values from the learnt SVM models are used as the mid-level features and the feature dimension is further reduced by using the SILD method [19]. Finally, the cosine similarity measure is employed for final face verification. Extensive experiments using two unconstrained face data sets demonstrate that our scheme outperforms most of the state-of-the-art methods.

Acknowledgment

The authors would like to thank the Associate Editor and the Reviewers for their valuable comments and suggestions.

References

[1] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.

[2] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, Dec. 2003.

[3] W. Y. Zhao, R. Chellappa, P. J. Phillips, and A. P. Rosenfeld, "Face recognition: A literature survey," *ACM Comput. Surveys*, vol. 35, no. 4, pp. 399–458, 2003.

[4] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Dept. Comput. Sci., Univ. Massachusetts, Amherst, MA, USA, Tech. Rep. 49, 2007.

[5] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Proc. Real-Life Images Workshop Eur. Conf. Comput. Vis.*, Oct. 2008, pp. 1–14.

[6] C. Sanderson and B. C. Lovell, "Multi-region probabilistic histograms for robust and scalable identity inference," in *Proc. Int. Conf. Biometrics*, 2009, pp. 199–208.

[7] Z. Cao, Q. Yin, X. Tang, and J. Sun, "Face recognition with learning-based descriptor," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2707–2714.

[8] N.-S. Vu and A. Caplier, "Face recognition with patterns of oriented edge magnitudes," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 313–326.

[9] N. Pinto and D. Cox, "Beyond simple features: A large-scale feature search approach to unconstrained face recognition," in *Proc. Int. Conf. Autom. Face Gesture Recognit.*, 2011, pp. 8–15.

[10] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep.–Oct. 2009, pp. 365–372.

[11] L. Wolf, T. Hassner, and Y. Taigman, "Similarity scores based on background samples," in *Proc. Asian Conf. Comput. Vis.*, 2009, pp. 88–97.

[12] L. Wolf, T. Hassner, and Y. Taigman, "The one-shot similarity kernel," in *Proc. Int. Conf. Comput. Vis.*, 2009, pp. 897–902.

[13] Y. Taigman, L. Wolf, and T. Hassner, "Multiple one-shots for utilizing class label information," in *Proc. Brit. Mach. Vis. Conf.*, 2009, pp. 1–12.

[14] E. Nowak and F. Jurie, "Learning visual similarity measures for comparing never seen objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[15] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? Metric learning approaches for face identification," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep.–Oct. 2009, pp. 498–505.

[16] H. V. Nguyen and L. Bai, "Cosine similarity metric learning for face verification," in *Proc. Asian Conf. Comput. Vis.*, 2010, pp. 709–720.

[17] Q. Yin, X. Tang, and J. Sun, "An associate-predict model for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 497–504.

[18] Z. Qiao, L. Zhou, and J. Z. Huang, "Sparse linear discriminant analysis with applications to high dimensional low sample size data," *Int. J. Appl. Math.*, vol. 39, no. 1, pp. 48–60, 2009.

[19] M. Kan, S. Shan, D. Xu, and X. Chen, "Side-information based linear discriminant analysis for face recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2011, pp. 1–12.

[20] H. Wang, S. Yan, D. Xu, X. Tang, and T. Huang, "Trace ratio vs. ratio trace for dimensionality reduction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[21] Y. Jia, F. Nie, and C. Zhang, "Trace ratio problem revisited," *IEEE Trans. Neural Netw.*, vol. 20, no. 4, pp. 729–735, Apr. 2009.

[22] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Stat.*, vol. 39, no. 4, pp. 407–499, 2004.

[23] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 7, pp. 711–720, Jul. 1997.

[24] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. Int. Conf. Mach. Learn.*, 1999, pp. 200–209.

[25] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, pp. 2399–2434, Nov. 2006.

[26] X. Zhu, "Semi-supervised learning literature survey," Dept. Comput. Sci., Univ. Wisconsin-Madison, Tech. Rep. 1530, 2005.

[27] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 529–534.

[28] L. Wiskott, J.-M. Fellous, N. Kruger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 775–779, Jul. 1997.

[29] Y. Su, S. Shan, X. Chen, and W. Gao, "Hierarchical ensemble of global and local classifiers for face recognition," *IEEE Trans. Image Process.*, vol. 18, no. 8, pp. 1885–1896, Aug. 2009.