Int J Comput Vis (2017) 123:479–498
DOI 10.1007/s11263-017-1000-3

Empowering Simple Binary Classifiers for Image Set Based Face Recognition

Munawar Hayat[1] · **Salman H. Khan**[2,3] · **Mohammed Bennamoun**[4]

Received: 6 October 2015 / Accepted: 10 February 2017 / Published online: 28 February 2017

Abstract Face recognition from image sets has numerous real-life applications including recognition from security and surveillance systems, multi-view camera networks and personal albums. An image set is an unordered collection of images (e.g., video frames, images acquired over long term observations and personal albums) which exhibits a wide range of appearance variations. The main focus of the previously developed methods has therefore been to find a suitable representation to optimally model these variations. This paper argues that such a representation could not necessarily encode all of the information contained in the set. The paper, therefore, suggests a different approach which does not resort to a single representation of an image set. Instead, the images of the set are retained in their original form and an efficient classification strategy is developed which extends well-known simple binary classifiers for the task of multi-class image set classification. Unlike existing binary to multi-class extension strategies, which require multiple binary classifiers to be trained over a large number of images, the proposed approach is efficient since it trains only few binary classifiers on very few images. Extensive experiments and comparisons with existing methods show that the proposed approach achieves state of the art performance for image set classification based face and object recognition on a number of challenging datasets.

Keywords Image set classification · Binary to multi-class classification · Video based face recognition · Object recognition

Communicated by K. Kise.

✉ Munawar Hayat
munawar.hayat@canberra.edu.au

Salman H. Khan
salman.khan@data61.csiro.au

Mohammed Bennamoun
mohammed.bennamoun@uwa.edu.au

[1] Human-Centered Technology Research Centre, University of Canberra, Bruce, ACT 2617, Australia

[2] Data61, Commonwealth Scientific and Industrial Research Organization (CSIRO), Canberra, ACT, Australia

[3] College of Engineering & Computer Science, Australian National University, Canberra, ACT, Australia

[4] School of Computer Science and Software Engineering, University of Western Australia, Crawley, WA 6009, Australia

1 Introduction

Owing to a wide range of potential applications, face recognition has been a research problem of significant importance in the area of computer vision and pattern recognition. Most of the effort in this regard has been tailored towards the classification from single images, that is, given a single query image, we are required to find its best match in a gallery of images. However, for many real-world applications (e.g., recognition from surveillance videos, multi-view camera networks and personal albums), multiple images of a person are readily available and need to be explored for classification. Face recognition from these multiple images is commonly studied under the framework of 'image set classification' and has attained significant research attention in the recent years (Kim et al. 2007; Wang et al. 2008; Wang and Chen 2009; Cevikalp and Triggs 2010; Harandi et al. 2011; Hu et al. 2012; Wang et al. 2012; Yang et al. 2013; Ortiz et al. 2013; Zhu et al. 2013; Hayat et al. 2014).

Compared with single image based classification, image set classification is more promising, since images in a set provide richer information due to wide range of appear-

ance variations caused by changing illumination conditions, head pose variations, expression deformations and occlusions. Although image set classification provides a plenitude of data of the same object under different variations, it simultaneously introduces many challenges e.g., how to make an effective use of this data. The major focus of existing image set classification methods has therefore been to find a suitable representation which can effectively model the appearance variations in an image set. For example, the methods in Kim et al. (2007), Yamaguchi et al. (1998), Oja (1983), Wang et al. (2008), Wang and Chen (2009), Hayat et al. (2013) and Hayat and Bennamoun (2014) use subspaces to model image sets, and set representative exemplars (generated from affine hull/convex hull) are used in Cevikalp and Triggs (2010) and Hu et al. (2012) for image set representations. The mean of the set images is used as part of image set representation in Ortiz et al. (2013), Hu et al. (2012) and Lu et al. (2013) and image sets are represented as a point on a manifold geometry in Wang et al. (2012) and Harandi et al. (2011). The main motivation behind a single entity representation of image sets (e.g., subspace, exemplar image, mean, a point on the manifold) is to achieve compactness and computational efficiency. However, these representations do not necessarily encode all of the useful information contained in the images of the image set (as explained in detail in Sect. 2). In this paper, we take a different approach which does not represent an image set by a single entity. We instead retain all the images of the image set in their original form and design an efficient classification framework to effectively deal with the plenitude of the data involved.

The proposed image set classification framework is built on well-developed learning algorithms. Although, these algorithms are originally designed for classification from single images, we demonstrate that they can be tailored for image set classification, by first individually classifying the images of a query set followed by an appropriate voting strategy (see Sect. 4.2). However, due to the plenitude of the data involved in the case of image set classification, a straight forward extension of these algorithms (from single image to image set classification) would be computationally burdensome. Specifically, since most of the popular learning algorithms (e.g., Support Vector Machines, AdaBoost, linear regression, logistic regression and decision tree algorithms) are inherently binary classifiers, their extension to a multi-class classification problem (such as image set classification) requires the training of multiple binary classifiers. One-vs-one and one-vs-rest are the two most commonly adopted strategies for this purpose. For a k-class classification problem, $\frac{k(k-1)}{2}$ and k binary classifiers are respectively trained for one-vs-one and one-vs-rest strategies. Although, one-vs-rest trains comparatively fewer classifiers, it still requires images from all classes to train each binary classifier. Adopting either of the well-known one-vs-one or one-vs-rest strategies for image set classification would therefore require a lot of computational effort, since either the number of images involved is quite large or a fairly large number of binary classifiers have to be trained.

The proposed framework in this paper trains a very small number of binary classifiers (mostly one or a maximum of five) on a very small fraction of images for the task of multi-class image set classification. The framework (see block diagram in Fig. 1) first splits the training images from all classes into two sets $\mathcal{D}_1$ and $\mathcal{D}_2$. The division is done such that $\mathcal{D}_1$ contains uniformly randomly sampled images from all classes with the total number of images in $\mathcal{D}_1$ being comparable to the number of images of the query image set. $\mathcal{D}_2$ contains all training images except the ones in $\mathcal{D}_1$. Next, a linear binary classifier is trained to optimally separate images of the query set from $\mathcal{D}_1$. Note that $\mathcal{D}_1$ has some images which belong to the class of the query set. However, since these images are very few in number, the classifier treats them as outliers. The trained classifier therefore learns to discriminate the class of the query set from all the other classes. Next, the learned classifier is evaluated on the images of $\mathcal{D}_2$. The images of $\mathcal{D}_2$ which are classified to belong to the images of the query set are of particular interest. Knowing the original class labels of these training images, we construct a histogram which is then used to decide about the class of the query set. A detailed description of the proposed framework is presented in Sect. 3 along with an illustration using a toy example in Fig. 3.

The main strengths of the proposed method are as follows. (1) A new strategy is introduced to extend any binary classifier for multi-class image set classification. Compared with the existing binary to multi-class strategies (e.g., one-vs-one and one-vs-rest), the proposed approach is computationally efficient to train. It only requires the training of a fixed number of binary classifiers (1–5 compared with k or $\frac{k(k-1)}{2}$) using a small number of images. (2) Along with the predicted class label of the query image set, the proposed method gives a confidence level of its prediction. This information is very useful and can be used as an indication of a potential miss-classification. The prior knowledge of a query image set being miss-classified allows for the potential use of another binary classifier. The proposed method can therefore accommodate the fusion of information from different types of binary classifiers before declaring the final class label of the query image set. (3) The proposed method is easily scalable to new classes. Unlike many existing image set classification methods, the computational complexity of the proposed method is not affected much by the addition of new classes in the gallery (see Sect. 4.2). Some of the existing methods would require retraining on the complete dataset (when new classes are enrolled), whereas, the proposed method requires no additional training and can efficiently discriminate the

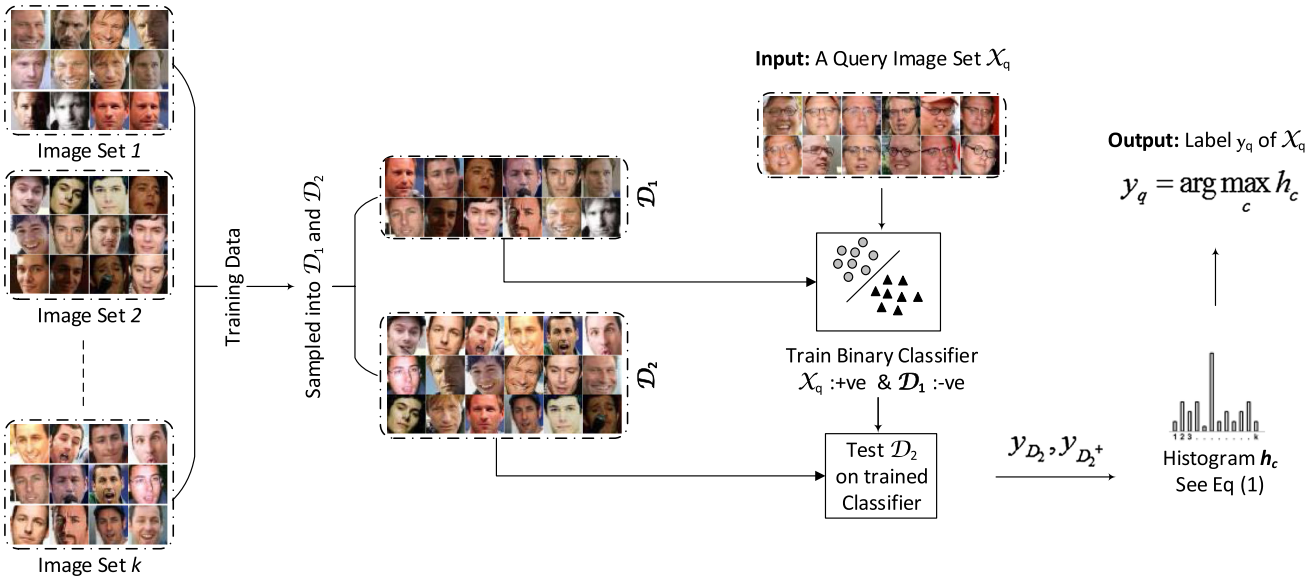

Fig. 1 Block diagram of the proposed method. The training data is divided into two sets $\mathcal{D}_1$ and $\mathcal{D}_2$. $\mathcal{D}_1$ contains uniformly randomly sampled images from all classes such that the size of $\mathcal{D}_1$ is comparable to the size of the query image set $\mathcal{X}_q$. A binary classifier is trained, with images of $\mathcal{X}_q$ (labelled $+1$) and $\mathcal{D}_1$ (labelled -1). The classifier is then tested on the images of $\mathcal{D}_2$. Knowing the class labels of images of $\mathcal{D}_2$ which are classified $+1$, we formulate a histogram (see Eq. 1), which is then used to decide about the class of $\mathcal{X}_q$. See a toy example in Fig. 3 for illustration

query class from other classes using a fixed number of binary classifiers (Sect. 4.7).

A preliminary version of our method appeared in Hayat et al. (2014). This paper extends (Hayat et al. 2014) in the following manners. (1) We encode a facial image in terms of the activations of a trained deep convolutional neural network. Compared with a shallow representation, the proposed learned feature representation proves to be more effective in discriminating images of different individuals (Sect. 3.1). (2) In order to further enhance the effectiveness of the proposed method, we propose three different sampling strategies. One of the proposed strategies also takes into consideration the head pose information of facial images which results in an overall improved performance of the method (Sect. 3.4). (3) We propose an extension of our method for the task of still to video face recognition which is an important and challenging real-life problem with numerous applications to security and surveillance systems (Sect. 4.4). (4) The efficacy of the proposed method is demonstrated through extensive experiments on four additional unconstrained real-life datasets (Sect. 4). We further extend our experimental evaluations by presenting a quantitative robustness analysis of different aspects of the proposed method (Sect. 4.5).

2 Related Work

The main focus of the existing image set classification methods is to find a suitable representation which can effectively model the appearance variations within an image set. Two different types of approaches have been previously developed for this purpose. The **first** approach models the variations within the images of a set through a statistical distribution and uses a measure such as KL-divergence to compare two sets. The methods based on this approach are called parametric model-based methods (Arandjelovic et al. 2005; Shakhnarovich et al. 2002). One of their major limitation is their reliance on a very strong assumption about the existence of a statistical correlation between image sets. The **second** approach for image set representation avoids such assumptions. The methods based on this approach are called non-parametric model-based methods (Kim et al. 2007; Wang et al. 2008; Wang and Chen 2009; Cevikalp and Triggs 2010; Harandi et al. 2011; Hu et al. 2012; Wang et al. 2012; Yang et al. 2013; Ortiz et al. 2013; Zhu et al. 2013; Hayat et al. 2014; Uzair et al. 2013) and have shown to give a superior performance compared with the parametric model-based methods. A brief overview of the non-parametric model-based methods is given below.

Subspaces have been very commonly used by the non-parametric methods to represent image sets. Examples include image sets represented by linear subspaces (Kim et al. 2007; Yamaguchi et al. 1998), orthogonal subspaces (Oja 1983) and a combination of linear subspaces (Wang et al. 2008; Wang and Chen 2009). Principal angles are then used to compare subspaces. A drawback of these methods is that they represent image sets of different sizes by a subspace of the same dimension. These methods cannot

therefore uniformly capture the critical information from image sets with different set lengths. Specifically, for sets with a larger number of images and diverse appearance variations, the subspace-based methods cannot accommodate all the information contained in the images. Image sets can also be represented by their geometric structures i.e., affine hull or convex hull models. For example, Affine Hull Image Set Distance (AHISD) (Cevikalp and Triggs 2010) and Sparse Approximated Nearest Points (SANP) (Hu et al. 2012) use affine hull, whereas Convex Hull Image Set Distance (CHISD) (Cevikalp and Triggs 2010) uses the convex hull of the images to model an image set. A set-to-set distance is then determined in terms of the Euclidean distance between the set representative exemplars which are generated from the corresponding geometric structures. Although these methods have shown to produce a promising performance, they are prone to outliers and are computationally expensive (since they require a direct one-to-one comparison of the query set with all sets in the gallery). Some of the non-parametric model-based methods represent an image set as a point on a certain manifold geometry e.g., Grassmannian manifold (Wang and Chen 2009; Harandi et al. 2011) and Lie group of Riemannian manifold (Wang et al. 2012). The mean of the set images has also been used either solely or as a part of image set representation in Ortiz et al. (2013), Hu et al. (2012) and Lu et al. (2013).

In this paper, we argue that a single entity (e.g., a subspace, a point on a manifold, or an exemplar generated from a geometric structure) for image set representation can be suboptimal and could result in the loss of information from the images of the set. For example, for image sets represented by a subspace, the amount of the retained information depends on the selected dimensions of the subspace. Similarly, generating representative exemplars from geometric structures could result in exemplars which are practically non-existent and are very different from the original images of the set. We, therefore, take a different approach which does not require any compact image set representation. Instead, the images are retained in their original form and a novel classification concept is proposed which incorporates well-developed learning algorithms to optimally discriminate the class of the query image set from all other classes. A detailed description of the proposed framework is presented next.

3 Proposed Method

Our proposed method first encodes raw face images in terms of the activations of a trained Convolutional Neural Network (CNN) (Sect. 3.1). The encoded face images are then used by the proposed image set classification algorithm, whose detailed description is presented in Sect. 3.2. Two important components of our proposed algorithm (*choice of the binary classifiers* and *sampling strategies*) are further elaborated in detail in Sects. 3.3 and 3.4, respectively. The proposed image set classification algorithm is then finally illustrated with the help of a toy example in Sect. 3.5.

3.1 Convolutional Feature Encoding

We are interested in mapping raw face images to a discriminative feature space where faces of different persons are easily separable. For this purpose, instead of using shallow or local feature representations (as in Hayat et al. 2014), we represent face images in terms of activations of a trained deep Convolutional Neural Network (CNN) model. Learned representations based on CNNs have significantly outperformed hand-crafted representations on nearly all major computer vision tasks (Chatfield et al. 2014; Jia et al. 2014; An et al. 2015; Khan et al. 2014). To this end, we adapt the parameters of AlexNet (Krizhevsky et al. 2012) (originally trained on 1.2 million images of 1000 object classes) for facial images. AlexNet consists of 5 convolutional and 3 fully-connected layers. In order to adapt the parameters of the network for facial images, we first encode faces of BU4DFE dataset (Yin et al. 2008) in terms of the activations of last convolutional layer. These encoded faces are then used as input to fine-tune the parameters of the three fully connected layers after changing the number of neurons in the last layer from 1000 (object categories in the ILSVRC Russakovsky et al. 2015) to 100 (number of subjects in the BU4DFE dataset). After learning the parameters of the fully connected part of the network, we append it back to the convolutional part, and fine-tune the complete network for facial images of BU4DFE dataset. Once the network parameters have been adapted, we feed the raw face images to the network's input layer after mean normalization. The processed output from the first fully connected layer of the network is considered to be our convolutional feature representation of the input face images. Apart from representing images in terms of the activations of AlexNet adapted for facial images of BU4DFE dataset, we also explore their representation in terms of activations of VGG-Face CNN model (Parkhi et al. 2015) which is specifically trained on 2.6 million facial images of 2, 622 subjects. A performance comparison of different feature encoding methods is presented in Sect. 4.6ii.

3.2 Image Set Classification Algorithm

3.2.1 Problem Description

For k classes of a training data, we are given k image sets $\mathcal{X}_1, \mathcal{X}_2, \ldots \mathcal{X}_k$ and their corresponding class labels $y_c \in [1, 2, \ldots k]$. An image set $\mathcal{X}_c = \{\mathbf{x}^{(t)} | y^{(t)} = c; t = 1, 2, \ldots N_c\}$ contains all N_c training images $\mathbf{x}^{(t)}$ belonging to class c. Note that for training data with multiple image

sets per class, we combine the images from all sets into a single set. During classification, we are given a query image set $\mathcal{X}_q = \{\mathbf{x}^{(t)}\}_{t=1}^{N_q}$, and the task is to find the class label y_q of $\mathcal{X}_q$.

The proposed image set classification algorithm is summarized in Algorithm 1. The details are presented below.

1. After encoding all the face images in terms of their convolutional activations, the images from all training sets are gathered into a single set $\mathcal{D} = \{\mathcal{X}_1, \mathcal{X}_2, \ldots \mathcal{X}_k\}$. Next, $\mathcal{D}$ is divided into two sets: $\mathcal{D}_1$ and $\mathcal{D}_2$ by adopting one of the sampling strategies described in Sect. 3.4. The division is done such that $\mathcal{D}_1$ contains an equal representation of images from all classes of the training data and the total number of images in $\mathcal{D}_1$ is comparable to that of $\mathcal{X}_q$. The class label information of images in $\mathcal{D}_1$ and $\mathcal{D}_2$ is stored in sets $\mathbf{y}_{\mathcal{D}_1} = \{y^{(t)} \in [1, 2, \ldots k], t = 1, 2, \ldots N_{\mathcal{D}_1}\}$ and $\mathbf{y}_{\mathcal{D}_2} = \{y^{(t)} \in [1, 2, \ldots k], t = 1, 2, \ldots N_{\mathcal{D}_2}\}$ respectively.
2. Next, we train a binary classifier C_1. Training is done on the images of $\mathcal{X}_q$ and $\mathcal{D}_1$. All images in $\mathcal{X}_q$ are labelled $+1$, while the images in $\mathcal{D}_1$ are labelled -1. Since images from all classes are present in $\mathcal{D}_1$, the classifier learns to separate images of $\mathcal{X}_q$ from the images of the other classes. Note that $\mathcal{D}_1$ does have a small number of images from the same class as of $\mathcal{X}_q$. However, since these images are very few, the binary classifier treats them as outliers and learns to discriminate the class of the query image set from all other classes (Sect. 4.5ii).
3. The trained classifier C_1 is then tested on the images of $\mathcal{D}_2$. The images in $\mathcal{D}_2$ classified as $+1$ (same as images of $\mathcal{X}_q$) are of interest. Let $\mathbf{y}_{\mathcal{D}_2^+} \subset \mathbf{y}_{\mathcal{D}_2}$ contain the class labels of images of $\mathcal{D}_2$ classified $+1$ by the classifier C_1.
4. A normalized frequency histogram $\mathbf{h}$ of class labels in $\mathbf{y}_{\mathcal{D}_2^+}$ is computed. The cth value of the histogram, $\mathbf{h}_c$, is given by the percentage of the images of class c in $\mathcal{D}_2$ which are classified $+1$. Formally, $\mathbf{h}_c$ is given by the ratio of *the number of images of* $\mathcal{D}_2$ *belonging to class c and classified as +1* **to** *the total number of images of* $\mathcal{D}_2$ *belonging to class c*. This is given by,

$$\mathbf{h}_c = \frac{\sum\limits_{y^{(t)} \in \mathbf{y}_{\mathcal{D}_2^+}} \delta_c(y^{(t)})}{\sum\limits_{y^{(t)} \in \mathbf{y}_{\mathcal{D}_2}} \delta_c(y^{(t)})}, \text{where} \tag{1}$$

$$\delta_c(y^{(t)}) = \begin{cases} 1, & y^{(t)} = c \\ 0, & \text{otherwise.} \end{cases}$$

5. A class in $\mathcal{D}_2$ with most of its images classified as $+1$ can be predicted as the class of $\mathcal{X}_q$. The class label y_q of $\mathcal{X}_q$ is therefore given by,

$$y_q = \arg\max_c \mathbf{h}_c. \tag{2}$$

We can also get a confidence level d of our prediction of y_q. This is defined in terms of the difference between the maximum and the second maximum values of histogram $\mathbf{h}$,

$$d = \max_{c \in \{1 \cdots k\}} \mathbf{h}_c - \max_{c \in \{1 \cdots k\} \setminus y_q} \mathbf{h}_c. \tag{3}$$

We are more confident about our prediction if the predicted class is a 'clear winner'. In the case of closely competing classes, the confidence level of the prediction will be low.

6. We declare the class label of $\mathcal{X}_q$ (as in Eq. 2) provided that the confidence d is greater than a certain threshold. The value of the threshold is determined empirically by performing experiments on a validation set. Otherwise, if the confidence level d is less than the threshold, steps 1–5 are repeated, for different random samplings of images into $\mathcal{D}_1$ and $\mathcal{D}_2$. After every iteration, a mean histogram $\bar{\mathbf{h}}$ is computed using the histogram of that iteration and the previous iterations. The confidence level d is also computed after every iteration using,

$$d = \max_{c \in \{1 \cdots k\}} \bar{\mathbf{h}}_c - \max_{c \in \{1 \cdots k\} \setminus y_q} \bar{\mathbf{h}}_c. \tag{4}$$

Iterations are stopped if the confidence level d becomes greater than the threshold or after a maximum of five iterations. Performing more iterations enhances the robustness of the method (since different images are selected into $\mathcal{D}_1$ and $\mathcal{D}_2$ for every iteration) but at the cost of an increased computational effort. Our experiments revealed that a maximum of five iterations is a good trade-off between the robustness and the computational complexity (Sect. 4.6iii).
7. If the confidence level d (see Eq. 4) is greater than the threshold, we declare the class label of $\mathcal{X}_q$ as $y_q = \arg\max_c \bar{\mathbf{h}}_c$. Otherwise, if the confidence level is lower than the threshold, declaring the class label would highly likely result in a miss-classification. In which case, we use another binary classifier C_2. The procedure is repeated for a different binary classifier C_2. The decision about y_q is then made based on the confidence levels of C_1 and C_2. The prediction of the more confident classifier is considered as the final decision. The description regarding the choice of the binary classifiers C_1 and C_2 is given next.

3.3 The Choice of Binary Classifiers

The proposed framework requires a binary classifier to distinguish between images of $\mathcal{X}_q$ and $\mathcal{D}_1$. The choice of the binary classifier should be based on its ability to generalize

Algorithm 1 The proposed Image Set Classification algorithm

Input: Training image sets $\mathcal{X}_1, \mathcal{X}_2, \ldots \mathcal{X}_k$; Query image set $\mathcal{X}_q$; *threshold*

$\mathcal{D} \leftarrow \{\mathcal{X}_1, \mathcal{X}_2, \ldots \mathcal{X}_k\}$ ▷ $\mathcal{D}$: All training images

$\mathcal{D}_1 \leftarrow \bigcup_c \mathcal{D}_{1c}$ where $\mathcal{D}_{1c}$ is a random subset of $\mathcal{X}_c$

$\mathcal{D}_2 \leftarrow \mathcal{D} \setminus \mathcal{D}_1$ ▷ $\mathcal{D}$ is divided into $\mathcal{D}_1$ and $\mathcal{D}_2$

$C_1 \leftarrow train(\mathcal{D}_1, \mathcal{X}_q)$ ▷ $\mathcal{X}_q$ labeled $+1$ and $\mathcal{D}_1$ labeled -1

$\mathbf{l}_{\mathcal{D}_2} \leftarrow test(C_1, \mathcal{D}_2)$ ▷ Test $\mathcal{D}_2$ on classifier C_1. $\mathbf{l}_{\mathcal{D}_2}$: binary labels of $\mathcal{D}_2$ images

$\mathbf{y}_{\mathcal{D}_2^+} \leftarrow \mathbf{l}_{\mathcal{D}_2}, \mathbf{y}_{\mathcal{D}_2}$ ▷ labels of images in $\mathcal{D}_2$ classified $+1$

$\mathbf{h} \leftarrow \mathbf{y}_{\mathcal{D}_2^+}, \mathbf{y}_{\mathcal{D}_2}$ ▷ Normalized histogram, see Eq 1

$d \leftarrow \mathbf{h}$ ▷ Confidence level, see Eq. 3

if $d > threshold$ **then**

  $y_q \leftarrow \arg\max_c \mathbf{h}_c$

else

  repeat ▷ Repeat for different random selections in $\mathcal{D}_1$ and $\mathcal{D}_2$

    $d, \bar{\mathbf{h}} \leftarrow$ Repeat steps 2-8

  until $d \geq threshold$ or repeated 5 times

  if $d > threshold$ **then**

    $y_q \leftarrow \arg\max_c \bar{\mathbf{h}}_c$

  else

    $d, \bar{\mathbf{h}} \leftarrow$ Repeat steps 2-14 for another binary classifier C_2

    $\bar{\mathbf{h}} \leftarrow$ Consider one from C_1 and C_2 with higher d

    $y_q \leftarrow \arg\max_c \bar{\mathbf{h}}_c$

end if

2**end if**

Output: Label y_q of $\mathcal{X}_q$

well to unseen data during the testing phase. Moreover, since the binary classifier is being trained on images of $\mathcal{X}_q$ and $\mathcal{D}_1$ and that some images in $\mathcal{D}_1$ have the same class as of $\mathcal{X}_q$, the binary classifier should not overfit on the training data and treat these images as outliers. For these reasons, a Support Vector Machine (SVM) with a linear Kernel is deemed to be an appropriate choice. It is known to produce an excellent generalization to unknown test data and can effectively handle outliers.

Two classifiers (C_1 and C_2) are used by the proposed framework. C_1 is a linear SVM with **L2** regularization and **L2** loss function, while C_2 is a linear SVM with **L1** regularization and **L2** loss function (Fan et al. 2008). Specifically, given a set of training example-label pairs $\left(\mathbf{x}^{(t)}, y^{(t)}\right)$, $y^{(t)} \in \{+1, -1\}$, C_1 solves the following optimization problem,

$$\min_{\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_t \left(\max\left(0, 1 - y^{(t)}\mathbf{w}^T\mathbf{x}^{(t)}\right)\right)^2, \tag{5}$$

while, C_2 solves the following optimization problem,

$$\min_{\mathbf{w}} \quad |\mathbf{w}|_1 + C\sum_t \left(\max\left(0, 1 - y^{(t)}\mathbf{w}^T\mathbf{x}^{(t)}\right)\right)^2. \tag{6}$$

Here, $\mathbf{w}$ is the coefficient vector to be learned and $C > 0$ is the penalty parameter used for regularization. After the learning of the SVM parameter $\mathbf{w}$, classification is performed based on the value of $\mathbf{w}^T\mathbf{x}^{(t)}$. Note that the coefficient vector $\mathbf{w}$ learned by classifier C_2 (trained for the challenging examples) is sparse. Learning a sparse $\mathbf{w}$ for C_2 further enhances the generalization capability for the challenging cases. We have also evaluated other binary classifiers which include non-linear SVM with Radial Basis Function (RBF) and Chi-Square kernels and random decision forests (Sect. 4.6i).

3.4 Sampling Strategies

Given all the training image sets $\mathcal{X}_1, \mathcal{X}_2, \cdots \mathcal{X}_k$, we gather these images (of the training data) into a set $\mathcal{D}$. Next, the images in $\mathcal{D}$ are sampled into two subsets ($\mathcal{D}_1$ and $\mathcal{D}_2$) which are used by the proposed algorithm, as explained in Sect. 3.2. For the sampling of the images of $\mathcal{D}$ to generate $\mathcal{D}_1$ and $\mathcal{D}_2$, we introduce three different sampling strategies. The following two general rules of thumb have been followed for sampling: (1) the total number of images in $\mathcal{D}_1$ are kept comparable to the number of images of the query set $\mathcal{X}_q$. Since our proposed image set classification algorithm trains a binary classifier to discriminate between $\mathcal{D}_1$ and $\mathcal{X}_q$, a huge disparity between number of images in $\mathcal{D}_1$ and $\mathcal{X}_q$ could result in a trained binary classifier which is biased towards the majority class. (2) Images in the sampled set $\mathcal{D}_1$ have an equal representation (>0) from all the classes of the training data. The detailed description of the proposed sampling strategies follows next.

3.4.1 Uniform Random Sampling

Let $\mathcal{D}_{1c}$ be a randomly sampled subset of $\mathcal{X}_c$ with a set size $N_{\mathcal{D}_{1c}}$, where $N_{\mathcal{D}_{1c}} = \left\lceil \frac{N_q}{k} \right\rceil$, such that $N_{\mathcal{D}_{1c}} \neq 0$ in any case,

then the set $\mathcal{D}_1$ is formed by the union operation: $\mathcal{D}_1 = \bigcup_c \mathcal{D}_{1c}, c = 1, 2, \ldots k$. $\mathcal{D}_2$ is obtained by $\mathcal{D}_2 = \mathcal{D} \setminus \mathcal{D}_1$.

3.4.2 Bootstrapped Sampling

We first perform bootstrapping and sample a set $\mathcal{D}'$ from $\mathcal{D}$ such that $\mathcal{D}' \subset \mathcal{D}$ and $\left|\mathcal{D}'\right| = \lfloor 0.8\,|\mathcal{D}| \rfloor$. Images in $\mathcal{D}'$ are randomly picked from $\mathcal{D}$. $\mathcal{D}_1$ and $\mathcal{D}_2$ are then uniformly randomly sampled from $\mathcal{D}'$ by following the same procedure described in Sect. 3.4.1. Sampling from the bootstrapped set $\mathcal{D}'$ over multiple iterations gives a data augmentation effect which eventually introduces robustness and results in an improved performance of the proposed method (Sect. 4.6).

3.4.3 Pose-Based Sampling

During our experiments, a visual inspection of the challenging YouTube celebrities dataset (Sect. 4.1) revealed that many of the miss-classified query image sets had face images with a head pose (such as profile views) which is otherwise not very commonly present in other training images. For such cases, only those images in $\mathcal{D}_2$ with the same pose as those of images of $\mathcal{X}_q$ (irrespective of their classes) are classified as $+1$. Our proposed pose-based sampling strategy aims to address this issue. The basic intuition here is to first estimate the pose of the images, and use this pose information to assign images into $\mathcal{D}_1$ and $\mathcal{D}_2$. For example, if most of the images of $\mathcal{X}_q$ are in right profile views, our sampling of the training images into $\mathcal{D}_1$ and $\mathcal{D}_2$ should consider only images with the right profile views. This helps to overcome any bias in the classification introduced by the head pose during classification.

In this strategy, we first determine the pose group of the face images using the pose group approximation method (described next). We then sample $\mathcal{D}'$ from $\mathcal{D}$ such that $\mathcal{D}'$ has only those images from $\mathcal{D}$ whose pose group is similar to the images of the query image set $\mathcal{X}_q$. Images from $\mathcal{D}'$ are then uniformly randomly sampled into $\mathcal{D}_1$ and $\mathcal{D}_2$ by following the procedure explained in Sect. 3.4.1. Note that $\mathcal{D}'$ is supposed to have an equal representation of images from all training image sets. However, we might not necessarily have images with the same pose as those of $\mathcal{X}_q$ for all training sets. From such training sets, we select the images with the most similar poses into $\mathcal{D}'$. The employed pose group approximation method Hayat et al. (2015) is described next.

3.4.4 Pose Group Approximation

An image is said to belong to a pose group $g \in \{1, 2, \ldots G\}$, if its pose along the pitch direction (y-axis) is within $\theta_g \pm 15°$. For our purpose, we define $G = 5$ and $\theta = \left[-60, -30, 0, 30, 60\right]$. The process of pose group approximation has two steps: training and testing.

Training Let $X_g \in \mathbb{R}^{m \times n_g}$ contain n_g images $\mathbf{x}^{(t)} \in \mathbb{R}^m$ whose pose is within $\theta_g \pm 15°$. We automatically select these images from a Kinect data set (see Sect. 4.1). The pose of Kinect images can be determined by the random regression forest based method of Fanelli et al. (2011a). From X_g, we want to extract the directions of major data orientation (principal directions). To achieve that, we first subtract the mean image from X_g and compute its covariance matrix Σ_g as follows,

$$\bar{X}_g = X_g - \frac{1}{n_g} \sum_t \mathbf{x}^{(t)}, \tag{7}$$

$$\Sigma_g = \bar{X}_g \bar{X}_g^T. \tag{8}$$

The singular value decomposition of the covariance matrix Σ_g results in $\Sigma_g = U_g S_g V_g$. The component U_g contains the eigenvectors arranged in the descending order of their significance. From U_g, we select the top k eigenvectors corresponding to the k largest eigenvalues and use them as columns to construct a matrix $\mathcal{S}_g \in \mathbb{R}^{m \times k}$. $\mathcal{S}_g$ is therefore a subspace whose columns represent the predominant data structure in the images of X_g. Next, during the testing phase of our pose group approximation approach, $\mathcal{S}_g$ is used for a linear regression based classification strategy.

Testing The pose group $\mathcal{P}(\mathbf{x}^{(t)})$ of the image $\mathbf{x}^{(t)}$ is determined by,

$$\mathcal{P}(\mathbf{x}^{(t)}) = \arg\min_g \left\| \mathbf{x}^{(t)} - \tilde{\mathbf{x}}_g^{(t)} \right\|_2, \tag{9}$$

where $\tilde{\mathbf{x}}_g^{(t)}$ is linearly constructed from $\mathcal{S}_g$ as follows,

$$\tilde{\mathbf{x}}_g^{(t)} = \mathcal{S}_g \alpha_g^{(t)}. \tag{10}$$

The above equation has an analytical solution given by,

$$\alpha_g^{(t)} = (\mathcal{S}_g^T \mathcal{S}_g)^{-1} \mathcal{S}_g^T \mathbf{x}^{(t)}. \tag{11}$$

A few sample results of our pose group approximation method are presented in Fig. 2. The pose group $\mathcal{P}(\mathbf{x}^{(t)})$ of all the images of the training data as well as the images of the query set $\mathcal{X}_q$ is determined by following the procedure explained above. Next, we sample images from $\mathcal{D}$ into $\mathcal{D}'$ such that images in $\mathcal{D}'$ have the same pose as those of images of $\mathcal{X}_q$. We ensure the inclusion of an equal representation of all classes in $\mathcal{D}'$. In case of classes with no or very few images with the same pose as of $\mathcal{X}_q$, images with nearly similar poses are selected. After getting $\mathcal{D}'$, we sample $\mathcal{D}_1$ and $\mathcal{D}_2$ by following the same procedure as explained in Sect. 3.4.1.

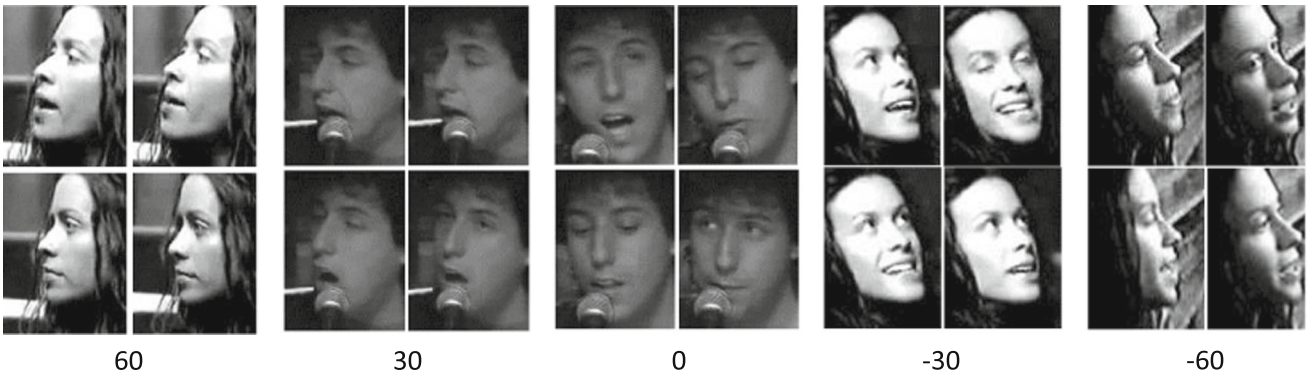

Fig. 2 Sample results of pose group approximation

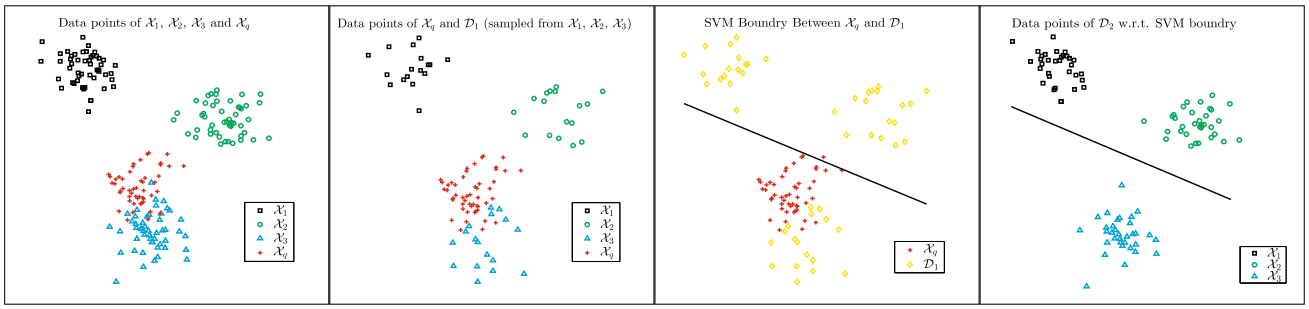

Fig. 3 Toy example to illustrate the proposed method. Consider a training set with three classes and the task is to find the class of $\mathcal{X}_q$ **(a)**. Data points from the three training image sets $\mathcal{X}_1$, $\mathcal{X}_2$, $\mathcal{X}_3$ and a query image set $\mathcal{X}_q$ are shown. **b** Data points from $\mathcal{X}_q$ and $\mathcal{D}_1$ (uniformly randomly sampled from $\mathcal{X}_1$, $\mathcal{X}_2$ and $\mathcal{X}_3$) are shown. **c** The learnt SVM boundary between $\mathcal{X}_q$ (labelled $+1$) and $\mathcal{D}_1$ (labelled -1). **d** The data points of $\mathcal{D}_2$ w.r.t. the learnt SVM boundary. Since the points of $\mathcal{X}_3$ in $\mathcal{D}_2$ lie on the same side of the boundary as the points of $\mathcal{X}_q$, the proposed method declares $\mathcal{X}_q$ to be from $\mathcal{X}_3$. Figure best seen in *color* (Color figure online)

3.5 Illustration with a Toy Example

The proposed image set classification algorithm is illustrated with the help of a toy example in Fig. 3. Let us consider a three class set classification problem in which we are given three training sets $\mathcal{X}_1$, $\mathcal{X}_2$, $\mathcal{X}_3$ and a query set $\mathcal{X}_q$. The data points of the training sets and the query set are shown in Fig. 3a. First, we form $\mathcal{D}_1$ by randomly sampling points from $\mathcal{X}_1$, $\mathcal{X}_2$ and $\mathcal{X}_3$. Fig. 3b shows the datapoints of $\mathcal{D}_1$ and $\mathcal{X}_q$. Next, a linear SVM is trained by labelling the datapoints of $\mathcal{X}_q$ as $+1$ and $\mathcal{D}_1$ as -1. Note that SVM (Fig. 3c) ignores the miss-labelled points (the points of $\mathcal{X}_3$ in $\mathcal{D}_1$) and treats them as outliers. Finally, we classify the data points of $\mathcal{D}_2$ from the learned SVM boundary. Figure 3d shows that the SVM labels the points of $\mathcal{X}_3$ in $\mathcal{D}_2$ as $+1$. The proposed algorithm therefore declares the class of $\mathcal{X}_3$ to be the class of $\mathcal{X}_q$.

4 Experiments

We perform experiments to evaluate the performance of the proposed method for two tasks (1) image set classification based face and object recognition, and (2) still to video imagery based face recognition. For image set classification based object recognition, experiments are performed on ETH-80 dataset (Leibe and Schiele 2003) while Honda/UCSD (Lee et al. 2003), CMU Mobo (Gross and Shi 2001), YouTube celebrities (Kim et al. 2008), a composite RGB-D Kinect dataset (obtained by combining three Kinect datasets), PubFig (Kumar et al. 2009), COX (Huang et al. 2013) and FaceScrub (Ng and Winkler 2014) datasets are used for image set classification based face recognition. For still to video face recognition, the COX dataset is used. It should be noted that most of the previous image set classification methods have only been evaluated on Honda, CMU Mobo, ETH-80 and YouTube celebrities dataset. Amongst these datasets, only YouTube celebrities dataset is a real life dataset whereas Honda, CMU Mobo and ETH-80 are considered relatively easy since they are acquired in indoor lab environment under controlled conditions. Apart from the challenging Youtube celebrities dataset, this paper also presents a comparative performance evaluation of our method with the existing methods (Yamaguchi et al. 1998; Kim et al. 2007; Wang et al. 2008; Wang and Chen 2009;

Cevikalp and Triggs 2010; Harandi et al. 2011; Hu et al. 2012; Wang et al. 2012; Ortiz et al. 2013; Zhu et al. 2013; Yang et al. 2013; Hayat et al. 2014) on three additional real-life datasets collected under unconstrained conditions. These include PubFig, COX and FaceScrub datasets.

Below, we first give a brief description of each of these datasets followed by the adopted experimental protocols (Sect. 4.1). We then present a performance comparison of our proposed method with the baseline multi-class classification strategies (Sect. 4.2) followed by a comparison with the existing state of the art image set classification methods (Sect. 4.3). The performance analysis for still to video based face recognition is presented in Sect. 4.4. A quantitative robustness analysis of different aspects of the proposed method is presented in Sect. 4.5. Finally, an ablative study to asses the contributions and impact of different components of our proposed method towards its overall performance is presented in Sec 4.6. A comparison of the computational complexity of different methods is given in Sect. 4.7.

4.1 Datasets and Experimental Settings

The Honda/UCSD Dataset Lee et al. (2003) contains 59 video sequences (with 12 to 645 frames in each video) of 20 subjects. We use Viola and Jones face detection (Viola and Jones 2004) algorithm to extract faces from video frames. The extracted faces are then resized to 20×20. For our experiments, we consider each video sequence as an image set and follow an evaluation configuration similar to Lee et al. (2003). Specifically, 20 video sequences are used for training and the remaining 39 sequences are used for testing. Three separate experiments are performed by considering all frames of a video as an image set and limiting the total number of frames in an image set to 50 and 100 (to evaluate the robustness for fewer images in a set). Each experiment is repeated 10 times for different random selections of the training and testing image sets.

The CMU Mobo (Motion of Body) Dataset Gross and Shi (2001) contains a total of 96 video sequences of 24 subjects walking on a treadmill. The faces from the videos are extracted using Viola and Jones (2004) and resized to 40×40. Similar to Wang et al. (2012) and Hu et al. (2012), we consider each video as an image set and use one set per subject for training and the remaining sets for testing. For a consistency, experiments are repeated ten times for different training and testing sets.

YouTube Celebrities Kim et al. (2008) dataset contains 1910 videos of 47 celebrities. The dataset is collected from YouTube and the videos are acquired under real-life scenarios. The faces in the dataset exhibit, therefore, a wide range of diversity and appearance variations in the form of changing illumination conditions, different head pose rotations and expression variations. Since the resolution of the face images is very low, face detection by Viola and Jones (2004) fails for a significant number of frames for this dataset. We, therefore, use tracking (Ross et al. 2008) to extract faces. Specifically, knowing the location of the face window in the first frame (provided with the dataset), we use the method of Ross et al. (2008) to track the face region in the subsequent frames. The extracted face region is then resized to 30×30. In order to perform experiments, we treat the faces acquired from each video as an image set and adopt a five fold cross validation experimental setup similar to Wang et al. (2008), Wang and Chen (2009), Hu et al. (2012) and Wang et al. (2012). The complete dataset is divided into five equal folds with minimal overlap. Each fold has nine image sets per subject, three of which are used for training and the remaining six are used for testing.

Composite Kinect Dataset is achieved by combining three distinct Kinect datasets: CurtinFaces (Li et al. 2013), Biwi Kinect (Fanelli et al. 2011a) and an in-house dataset acquired in our laboratory. The number of subjects in each of these datasets is 52 (5000 RGB-D images), 20 (15,000 RGB-D images) and 48 (15000 RGB-D images) respectively. The random forest regression based classifier of Fanelli et al. (2011b) is used to detect faces from the Kinect acquired images. The images in the composite dataset have a large range of variations in the form of changing illumination conditions, head pose rotations, expression deformations, sun glass disguise, and occlusions by hand. For performance evaluation, we randomly divide RGB-D images of each subject into five uniform folds. Considering each fold as an image set, we select one set for training and the remaining sets for testing. The experiments are repeated five times for different selections of training and testing sets.

ETH-80 Object Dataset contains still RGB images of eight object categories. These include cars, cows, apples, dogs, cups, horses, pears and tomatoes. Each object category is further divided into ten subcategories such as different brands of cars or different breeds of dogs. Each subcategory contains images under 41 orientations. For our experiments, we use the 128×128 cropped images [1] and resize them to 32×32. We follow an experimental setup similar to Wang and Chen (2009), Kim et al. (2007) and Wang et al. (2012). Images of an object in a subcategory are considered as an image set. For each object, five subcategories are randomly selected for training and the remaining five are used for testing. Ten runs of experiments are performed for different random selections of the training and testing sets.

Public Figures Face Database (PubFig) Kumar et al. (2009) is a real-life dataset of 200 people collected from the internet. The images (static RGB) of the dataset have been acquired in uncontrolled situations without any user cooperation. The sample images of a subject in Fig. 4 illustrate the large variations in the images caused by pose, lighting, expressions, backgrounds and camera positions. For our experiments, we

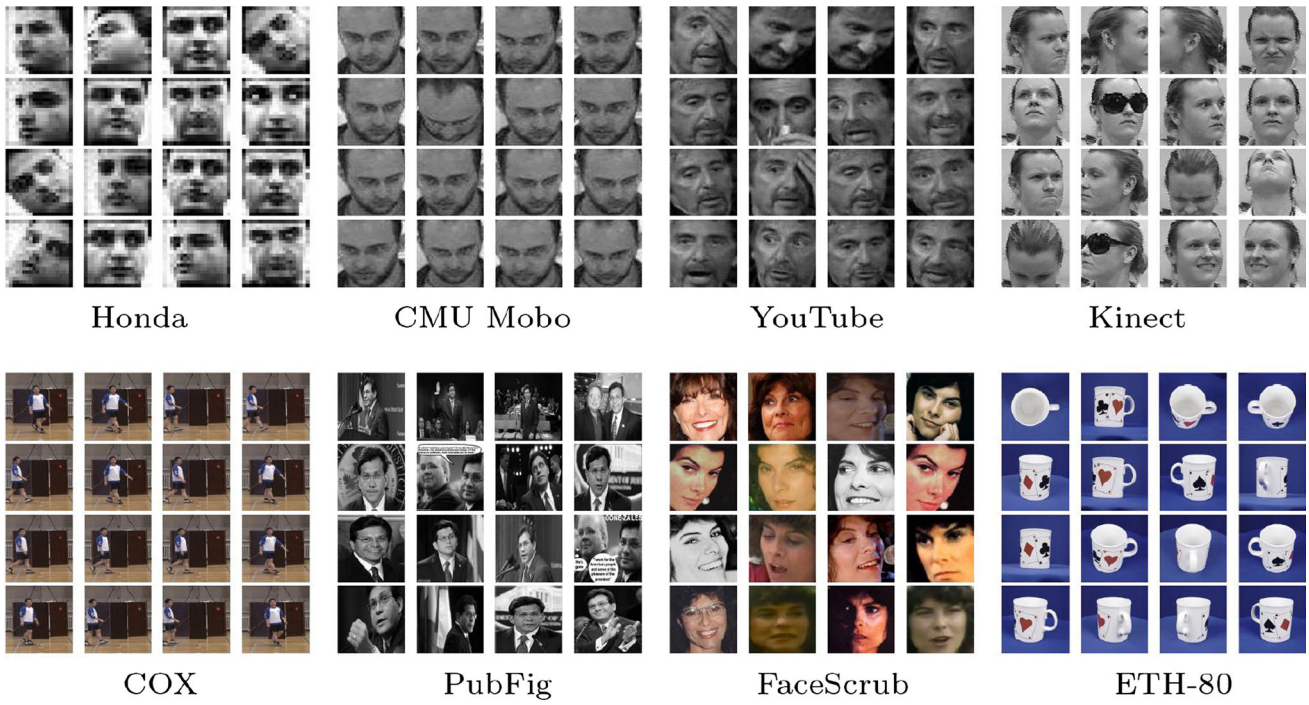

Fig. 4 Sample images from different datasets. Note the high intra class variations in the form of different head poses, illumination variations, expression deformations and occlusions

divide equally the images of each subject into three folds. Considering each fold as an image set, we use one of them for training and the remaining two are used for testing. Experiments are repeated five times for different random selections of images for the training and testing folds.

The COX (Huang et al. 2013) *Dataset* contains 1000 high resolution still images and 4000 uncontrolled low resolution video sequences of 1000 subjects. The videos have been captured inside a gymnasium with subjects walking naturally and without any restriction on expression and head orientation. The dataset contains four videos per subject. The face resolution, head orientation and lighting conditions in each video are significantly different from the others. Sample images of a subject from this dataset are shown in Fig. 4. For our image set classification experiments, we use the frames of each video as an image set and follow a leave-one-out strategy where one image set is held out for testing and remaining are used for training. For consistency, four runs of experiments are performed by swapping the training and testing image sets.

For still to video based face recognition experiments, we consider the high resolution still images (which were acquired with the full user cooperation) as our gallery. The low resolution images of the video sequence are used as the probe image set. Still to video based face recognition experiments are performed by following the standard evaluation protocol described in Huang et al. (2014). The still images and images from the video sequences of 300 individuals are randomly selected to learn a common embedding space for both the low and high resolution images using the technique in Sharma et al. (2012). The images of the remaining 700 individuals are used for testing. Experiments are repeated five times for different random shuffling of subjects between the training and testing sets. A common embedding space is learnt because the gallery and probe data possess very different visual characteristics *i. e.* the gallery contains good quality frontal face images acquired with full-user cooperation whereas the probes are low quality non-frontal images acquired without any cooperation.

FaceScrub Ng and Winkler (2014) is a large dataset of 530 (265 males and 265 females) celebrities and famous personalities. The dataset is collected from the internet and comprises a total of 107,818 RGB face images with approximately 200 images per person. Few sample images of a person in Fig 4 show the wide range of appearance variations in the images of an individual from the dataset. For our experiments, we divide the images of each person into ten folds. Considering each fold as an image set, we use one fold for training and the remaining are used for testing. Experiments are done five different times for a different random selection of images into each fold.

Following the evaluation configurations described above, we perform experiments and compare our method with the baseline methods, and current state of the art methods. A detailed analysis, extensive performance evaluations and comparisons are presented next.

Table 1 Performance comparison with the baseline methods

Methods	Honda	Mobo	YTC	Kinect
one-vs-one	92.1 ± 2.2	94.7 ± 2.0	67.7 ± 4.0	94.3 ± 3.5
one-vs-rest	94.6 ± 1.9	96.7 ± 1.6	68.4 ± 4.2	94.6 ± 3.3
This paper	100.0 ± 0.0	98.3 ± 0.7	77.4 ± 3.5	98.3 ± 1.7
	ETH	PubFig	COX	FS
one-vs-one	96.2 ± 2.9	87.4 ± 0.5	67.4 ± 11.5	80.1 ± 1.5
one-vs-rest	97.6 ± 1.5	89.1 ± 0.1	68.2 ± 11.4	80.3 ± 1.5
This paper	96.1 ± 1.8	98.6 ± 0.3	74.1 ± 10.2	91.5 ± 0.5

Average identification rates (percentage) of our method and two well-known multi-class classification strategies. See Table 2 for a comparison of the computational complexity

4.2 Comparison with the Baseline Methods

Linear SVM based one-vs-one and one-vs-rest multi-class classification strategies are used as baseline methods for comparison. Note that these baseline methods are suitable for classification from single images. For image set classification, we first individually classify every image of the query image set followed by a majority voting to decide about the class of the query image set. Experimental results in terms of average identification rates and standard deviations on all datasets are presented in Table 1. Note that for the Honda dataset, we perform three experiments i.e., by considering all frames of the video as an image set, then limiting the number of images in a set to 100 and 50 (see Sect. 4.3). Here, the results presented for the Honda/UCSD dataset are only for all frames of the videos considered as image sets. The results show that, amongst the compared baseline multi-class classification strategies, one-vs-rest performs slightly better than one-vs-one. Our method performs better than the baseline methods on all datasets except ETH-80. A possible explanation for a lower performance on the ETH-80 is that the proposed method trains a binary classifier on images of $\mathcal{X}_q$ and $\mathcal{D}_1$, which is then evaluated on $\mathcal{D}_2$. The set $\mathcal{D}_1$ contains $\left\lceil \frac{N_q}{k} \right\rceil$ images with same label as $\mathcal{X}_q$. For larger k, these images are few in number and do not affect training of the binary classifier. However, for smaller values of k (as is the case for ETH-80 dataset, $k = 8$) the proportion of these images is higher and causes slight performance degradation. A quantitative robustness analysis of the proposed method for different values of k is presented in Sect. 4.5iii.

Table 2 presents a comparison of the computational complexity in terms of the required number of binary classifiers and the number of images used to train each of these classifiers. One-vs-one trains $\frac{k(k-1)}{2}$ binary classifiers and uses images from two classes to train each classifier. Although the number of trained classifiers for one-vs-rest are comparatively less (k compared with $\frac{k(k-1)}{2}$), the number of images used to train each binary classifier is quite large (all images of the dataset are used). In comparison, our proposed method trains only few binary classifiers (a maximum of five for the challenging cases) and the number of images used for training is also small. A main difference of our method from baseline strategies is that it performs all computations at runtime.

4.3 Comparison with Existing Image Set Classification Methods

We present a comparison of our method with a number of recently proposed state of the art image set classification methods. The compared methods include Mutual Subspace Method (Yamaguchi et al. 1998), Discriminant Canonical Correlation Analysis (DCC) (Kim et al. 2007), Manifold-to-Manifold Distance (MMD) (Wang et al. 2008), Manifold Discriminant Analysis (MDA) (Wang and Chen 2009), the Linear version of the Affine Hull-based Image Set Distance (AHISD) (Cevikalp and Triggs 2010), the Convex Hull-based Image Set Distance (CHISD) (Cevikalp and Triggs 2010), Sparse Approximated Nearest Points (SANP) (Hu et al. 2012), Covariance Discriminative Learning (CDL) (Wang et al. 2012), Mean Sequence Sparse Representation Classification (MSSRC) (Ortiz et al. 2013), Set to Set Distance Metric Learning (SSDML) (Zhu et al. 2013), Regularized Nearest Points (RNP) (Yang et al. 2013) and Non-Linear Reconstruction Models (NLRM). We use the implementations provided by the respective authors for all methods. The parameters for all methods are optimized for best performance.

Specifically, for MSM, Principal Component Analysis (PCA) is applied to retain 90% of the total energy. For DCC, the dimensions of the embedding space are set to 100. The number of retained dimensions for a subspace are set to 10 (90% energy is preserved) and the corresponding 10 maximum canonical correlations are used to compute set-set similarity. For datasets with one training set per class (Honda/UCSD, CMU, Kinect, PubFig, COX and FaceScrub), we randomly divide the training set into two subsets to construct the within class sets as in Kim et al.

Table 2 Complexity analysis

Method	Total binary classifiers	Images to train each classifier
One-vs-one	$\frac{k(k-1)}{2}$ {1081}	$2N_c$ {600}
One-vs-rest	k {47}	$\sum_{c=1}^{k} N_c$ {14000}
This paper	$1-5$	$2N_q$ {200}

The proposed method trains just few binary classifiers and the number of images used for training is very small. The typical parameters values for YouTube celebrities dataset are given in brackets

Table 3 Performance comparison on Honda/UCSD dataset

	MSM	DCC	MMD	MDA	AHISD	CHISD	SANP
All	88.2 ± 3.8	92.5 ± 2.2	92.0 ± 2.2	94.3 ± 3.3	91.2 ± 1.7	93.6 ± 1.6	95.1 ± 3.0
100	85.6 ± 4.3	89.2 ± 2.4	85.5 ± 2.1	91.7 ± 1.6	90.7 ± 3.2	91.0 ± 1.7	94.1 ± 3.2
50	83.0 ± 1.7	82.0 ± 3.3	83.1 ± 4.4	85.6 ± 5.8	89.8 ± 2.1	90.5 ± 2.0	91.9 ± 2.7
	CDL	MSSRC	SSDML	RNP	NLRM	This paper	
All	98.9 ± 1.3	97.9 ± 2.6	86.4 ± 3.6	95.9 ± 2.1	100.0 ± 0.0	100.0 ± 0.0	
100	96.2 ± 1.2	96.9 ± 1.3	84.3 ± 2.2	92.3 ± 3.2	100.0 ± 0.0	100.0 ± 0.0	
50	93.9 ± 2.2	94.3 ± 1.4	83.4 ± 1.7	90.2 ± 3.2	100.0 ± 0.0	100.0 ± 0.0	

Average identification rates (percentage) and standard deviations of different methods when tested on the Honda/UCSD dataset. These experiments are performed by considering all the frames of the video as an image set and limiting the set length to 100 and 50 frames. The results show that the proposed method does not only achieve the best performance but it also maintains consistency in its performance for reduced set lengths

(2007). The parameters for MMD and MDA are used from Wang et al. (2008) and Wang and Chen (2009) respectively. The number of connected nearest neighbours to compute the geodesic distance is either set to 12 or to the number of images in the smallest image set of the dataset. The ratio between the Euclidean distance and the geodesic distance is optimized for all data sets. In case of MMD, the distance is computed in terms of maximum canonical correlation. No parameter settings are required for AHISD. For CHISD, the same error penalty term ($C = 100$) as in Cevikalp and Triggs (2010) is used. For SANP, the same weight parameters as in Hu et al. (2012) are adopted for optimization. For GEDA, we set $k^{[cc]} = 1$, $k^{[proj]} = 100$ and $v = 3$ (the value of v is searched over a range of 1-10 for best performance). The number of eigenvectors r used to represent an image set is set to 9 and 6, respectively, for Mobo and YouTube celebrities and 10 for all other datasets. No parameter settings are required for CDL. For RNP (Yang et al. 2013), PCA is applied to preserve 90% of the energy and the same weight parameters as in Yang et al. (2013) are used. No parameter configurations are required for MSSRC and SSDML. For NLRM, we use majority voting and perform PCA to retain the dimensions of the embedded space to 400.

The experimental results, in terms of the average identification rates and standard deviations of the different methods on the Honda/UCSD dataset, are presented in Table 3. The proposed method achieves a perfect classification for all frames of the video sequence (considered as an image set) as well as when the total number of images in the set is reduced to 100 and 50. This proves that our method is robust w.r.t. the number of images in the set and it is suitable for real-life scenarios (where only a limited number of images are available in a set).

The average identification rates and standard deviations of the different methods when tested on CMU Mobo, YouTube Celebrities (YTC), Kinect, ETH-80, PubFig, COX and Face-Scrub (FS) datasets are summarized in Table 4. The results prove that the proposed method outperforms most of the existing methods on all datasets. The gain in performance is more significant for YTC, PubFig and FS datasets. Note that YTC, PubFig and FS are very challenging datasets since their images have been acquired in real life scenarios without any user cooperation. The proposed method achieves a relative performance boost of 8.4, 11.0 and 12.7% on YTC, PubFig and FS datasets, respectively. Another notable aspect of the proposed method is that it not only works for image set classification based face recognition but also achieves a very high identification rate of 96.1% for the task of image set classification based object recognition.

The performance of all methods is further analyzed in Figs. 5, 6 on four real-life datasets which include YTC, PubFig, COX and FS. Specifically, Cumulative Match Characteristics (CMC) and Receiver Operating Characteristics (ROC) curves for the top performing methods are presented

Table 4 Performance evaluation of all methods on different datasets

Methods	Mobo	YTC	Kinect	ETH	PubFig	COX	FS
MSM FG'98 Yamaguchi et al. (1998)	96.8 ± 2.0	50.2 ± 3.6	89.3 ± 4.1	75.5 ± 4.8	57.0 ± 2.6	26.4 ± 10.9	57.7 ± 0.9
DCC TPAMI'07 Kim et al. (2007)	88.9 ± 2.5	51.4 ± 5.0	92.5 ± 2.0	91.8 ± 3.7	34.9 ± 7.7	43.3 ± 12.1	57.9 ± 4.4
MMD CVPR'08 Wang et al. (2008)	92.5 ± 2.9	54.0 ± 3.7	93.9 ± 2.3	77.5 ± 5.0	36.2 ± 6.9	54.9 ± 10.3	72.5 ± 4.5
MDA CVPR'09 Wang and Chen (2009)	81.0 ± 12.3	55.1 ± 4.6	93.5 ± 3.6	77.3 ± 5.5	34.3 ± 6.4	73.1 ± 10.4	79.2 ± 3.5
AHISD CVPR'10 Cevikalp and Triggs (2010)	92.9 ± 2.1	61.5 ± 5.6	91.6 ± 2.2	78.6 ± 5.3	62.1 ± 2.0	64.1 ± 11.3	68.5 ± 1.1
CHISD CVPR'10 Cevikalp and Triggs (2010)	96.5 ± 1.2	60.4 ± 6.0	92.7 ± 1.9	79.5 ± 5.3	64.8 ± 2.1	63.1 ± 10.4	71.2 ± 1.2
GEDA CVPR'11 Harandi et al. (2011)	84.9 ± 3.2	52.5 ± 4.5	91.4 ± 6.3	79.5 ± 5.2	35.5 ± 26.2	53.2 ± 15.8	52.5 ± 3.5
SANP TPAMI'12 Hu et al. (2012)	97.6 ± 0.9	65.6 ± 5.6	93.8 ± 3.1	77.8 ± 7.3	80.4 ± 2.5	66.2 ± 13.4	69.7 ± 1.7
CDL CVPR'12 Wang et al. (2012)	90.0 ± 4.4	56.4 ± 5.3	94.6 ± 1.0	77.8 ± 4.2	51.1 ± 4.0	56.1 ± 16.3	66.1 ± 1.3
MSSRC CVPR'13 Ortiz et al. (2013)	97.5 ± 0.9	59.4 ± 5.7	95.5 ± 2.3	90.5 ± 3.1	85.6 ± 2.8	69.4 ± 15.7	81.2 ± 2.1
SSDML ICCV'13 Zhu et al. (2013)	95.1 ± 2.2	66.2 ± 5.2	86.9 ± 3.4	81.0 ± 6.6	88.8 ± 1.6	65.3 ± 10.8	75.4 ± 0.8
RNP FG'13 Yang et al. (2013)	96.1 ± 1.4	65.8 ± 5.4	96.2 ± 2.5	81.0 ± 3.2	88.6 ± 1.0	66.2 ± 12.8	73.6 ± 0.8
NLRM CVPR'14 Hayat et al. (2014)	97.9 ± 0.7	71.4 ± 4.8	98.1 ± 1.7	98.1 ± 1.7	88.6 ± 1.5	66.1 ± 14.7	66.0 ± 3.0
This paper	98.3 ± 0.7	77.4 ± 3.5	98.3 ± 1.7	96.1 ± 1.8	98.6 ± 0.3	74.1 ± 10.2	91.5 ± 0.5

Average identification rates (percentage) and standard deviations on CMU/Mobo, YouTube Celebrities (YTC), Kinect, ETH-80, PubFig, COX and FaceScurb (FS) datasets. The proposed method achieves the best performance on most of these datasets with a significant performance boost on YTC, PubFig and FS datasets

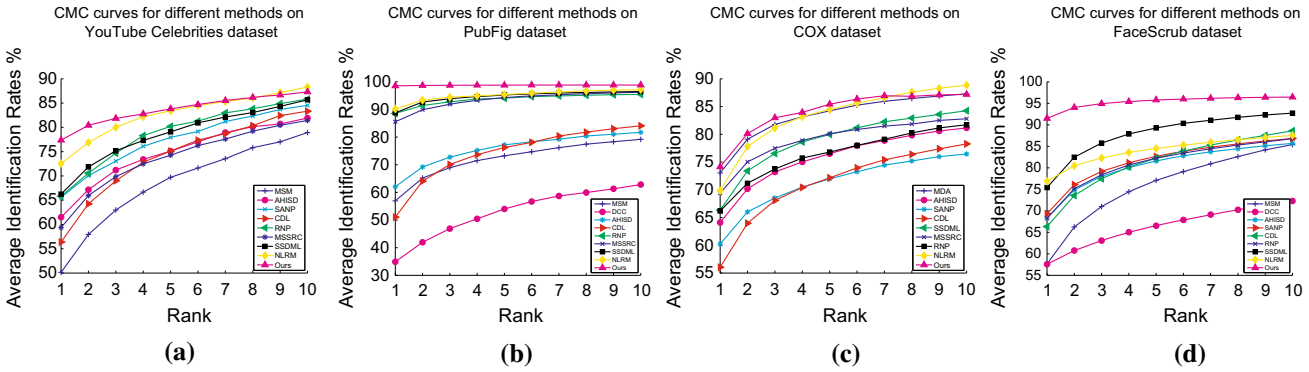

Fig. 5 Cumulative match characteristic (CMC) curves on YTC, PubFig, COX and FS datasets. Figure best seen in *colors* . **a** YTC, **b** PubFig, **c** COX, **d** FaceScrub (Color figure online)

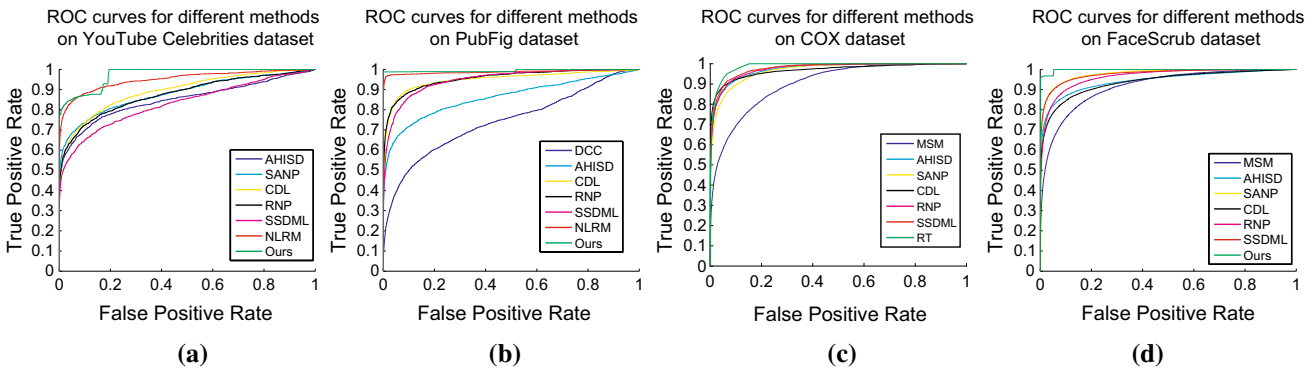

Fig. 6 Receiver operating characteristics (ROC) curves on YTC, PubFig, COX and FS datasets. Figure best seen in *colors*. **a** YTC, **b** PubFig, **c** COX, **d** FaceScrub (Color figure online)

in Figs. 5, 6 respectively. The results in Fig. 5 suggest that the proposed method consistently achieves the highest rank-1 to rank-10 identification rates for most of the evaluated datasets. ROC curves in Fig. 6 show that the proposed method outperforms all others. Equal error rates are shown in Fig. 7 to compare the verification performance of different methods on all datasets. The results show that the proposed method achieves superior performance by producing the lowest equal error rates compared with the existing methods on almost all of the evaluated datasets.

The state of the art performance of the proposed method is attributed to the fact that (unlike existing methods) it does not resort to a single entity representation (such as a subspace, the mean of set images or an exemplar image) for all images of the set. Any potential loss of information is therefore avoided by retaining the images of the set in their original form. Moreover, well-developed classification algorithms are efficiently incorporated within the proposed framework to optimally discriminate the class of the query image set from the remaining classes. Furthermore, since the proposed method provides a confidence level for its prediction, the classification decisions from multiple classifiers can be fused to enhance the overall performance of the method.

4.4 Still to Video Face Recognition

We also validate our proposed approach for still-to-video based face recognition which finds its usefulness in numerous real-life applications such as face recognition from surveillance cameras. The only modification required to adapt the proposed method to the case of still to video face recognition is to perform more iterations in steps 1–5 of the original algorithm. For this, we enforce an upper limit of 10 iterations. Table 5 compares our proposed method against a number of recent works, which can be adapted to the case of still to video based face recognition. These include the baseline Nearest Neighbour (NN) Classifier, Neighbourhood Component Analysis (NCA) (Goldberger et al. 2004), Information Theoretic Machine Learning (ITML) (Davis et al. 2007), Local Fisher Discriminant Analysis (LFDA) (Sugiyama 2007), Large Margin Nearest Neighbor (LMNN) (Weinberger and Saul 2009), Nearest Feature Classifiers (NFC) (Chien and Wu 2002), Hyperplane Distance Nearest Neighbor (HKNN) (Vincent and Bengio 2001), K-local Convex Distance Nearest Neighbors (CKNN), Mahalanobis Distance (MD), Point to Set Distance Metric Learning (PSDML) (Zhu et al. 2013) and Learning Euclidean to Riemannian Metric (LERM)

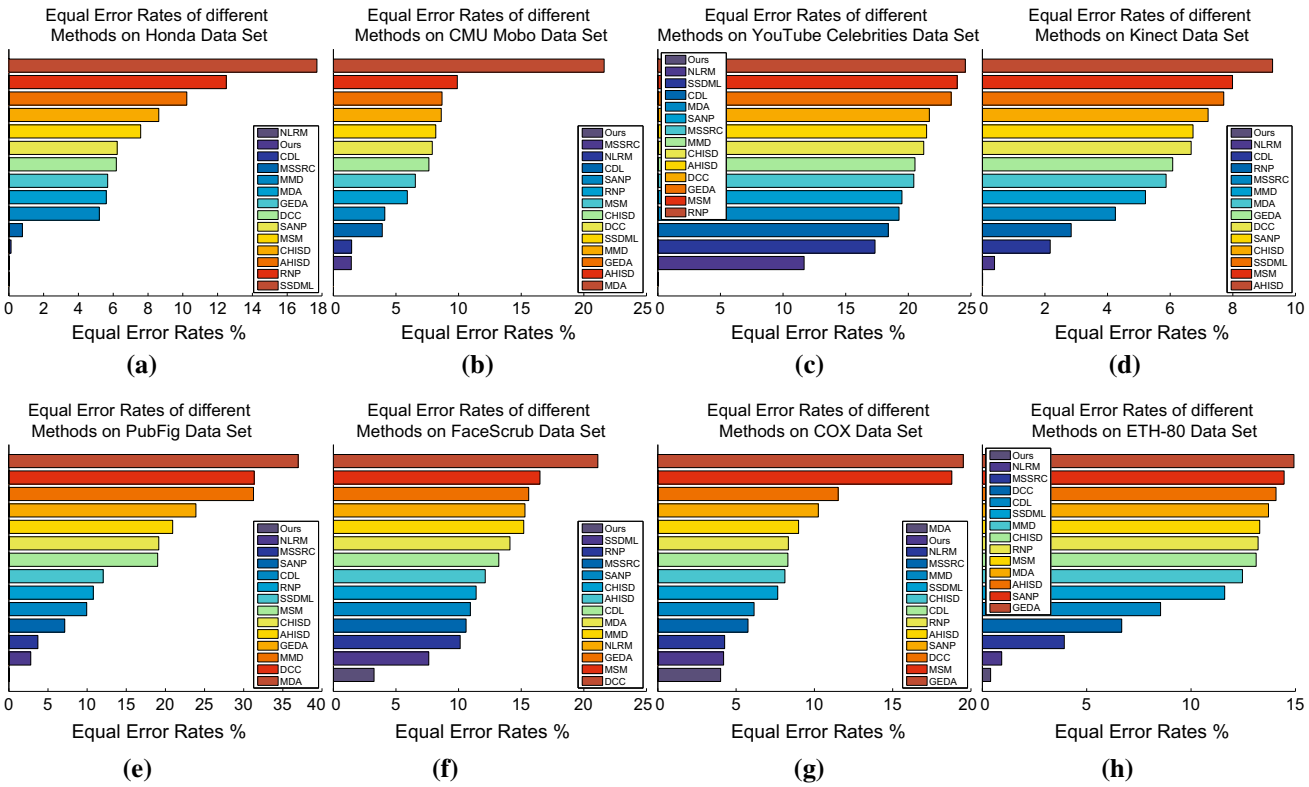

Fig. 7 Equal error rates (EERs) of different methods on all datasets. Figure best seen in *colors*. **a** Honda, **b** CMU, **c** YTC, **d** Kinect, **e** PubFig, **f** FaceScrub, **g** COX, **h** ETH-80 (Color figure online)

(Huang et al. 2013). Experiments are conducted using COX still-to-video dataset. The results in Table 5 illustrate the superior performance of our method and its suitability for the challenging and important problem of still to video based face recognition from surveillance imagery.

4.5 Robustness Analysis

In order to analyse the robustness of the proposed method with respect to its different aspects, we conduct quantitative experimental evaluations. In this regards, the following aspects are explored. (i) Number of images in the gallery and the probe sets (ii) number of images in sets $\mathcal{D}_1$ & $\mathcal{D}_2$, and (iii) number of enrolled subjects in the gallery. These experimental evaluations and the achieved results are discussed next.

(i) Size of Gallery and Probe Image Sets We perform experiments on YouTube celebrities dataset by enforcing an upper limit on number of images in the sets. Specifically, by keeping the size of the probe image sets fixed, we first gradually reduce the number of images in gallery sets from 250 to 8. We then keep the size of the gallery image sets fixed, and gradually decrease the size of the probe image sets. The achieved experimental results for reduced gallery and probe sets are presented in Fig 8a, b respectively. The results suggest that the performance of the proposed method is quite robust to the size of the probe image sets. Reducing the size of the probe image sets to as low as 8 images achieves a classification accuracy of 72.1% (compared to 77.4% for full size). Reducing the size of the gallery image set beyond 25 images, however, does cause a noticeable performance drop. The proposed method can still achieve a performance of 61.4% when the gallery set size is reduced to only 8 images.

(ii) Size of $\mathcal{D}_1$ and $\mathcal{D}_2$ The proposed method trains a binary classifier between images of $\mathcal{X}_q$ and $\mathcal{D}_1$ which is then evaluated on $\mathcal{D}_2$. $\mathcal{D}_1$ has $N_{\mathcal{D}_{1c}}=\left\lceil \frac{N_q}{k} \right\rceil$ uniformly sampled images from each class of the training data. It also contains $N_{\mathcal{D}_{1c}}$ miss-labelled images (which have the same label as $\mathcal{X}_q$). Increasing the size of $\mathcal{D}_1$ will decrease the size of $\mathcal{D}_2$ and also increase the number of miss-labelled images in $\mathcal{D}_1$. This will cause the performance to drop. In order to quantitatively evaluate the robustness of the proposed method against number of images in $\mathcal{D}_1$ and $\mathcal{D}_2$, we gradually increase the number of images sampled from each class of the training data to form $\mathcal{D}_1$ from $N_{\mathcal{D}_{1c}}$ to $mN_{\mathcal{D}_{1c}}$. Experimental results on YouTube celebrities dataset for $m = \{0.5, 1, 1.5, 2, 2.5, 3, 4, 5, 6, 7\}$ are presented in Table 6. The results show that the performance of the method only drops from 77.7 to 73.2% when there is a 14 fold increase (from $m = 0.5$ to $m = 7$) in the number of images in $\mathcal{D}_1$. A possible reason for this perfor-

Table 5 Still to video face recognition

Method	Accuracy	Method	Accuracy
NNC	11.5	CKNN Vincent and Bengio (2001)	9.4
NCA Goldberger et al. (2004)	42.8	MD	15.1
ITML Davis et al. (2007)	24.9	PSDML Zhu et al. (2013)	15.6
LFDA Sugiyama (2007)	29.2	HKNN Vincent and Bengio (2001)	7.0
LMNN Weinberger and Saul (2009)	40.8	LERM Huang et al. (2014)	48.8
NFC Chien and Wu (2002)	12.7	This paper	51.2

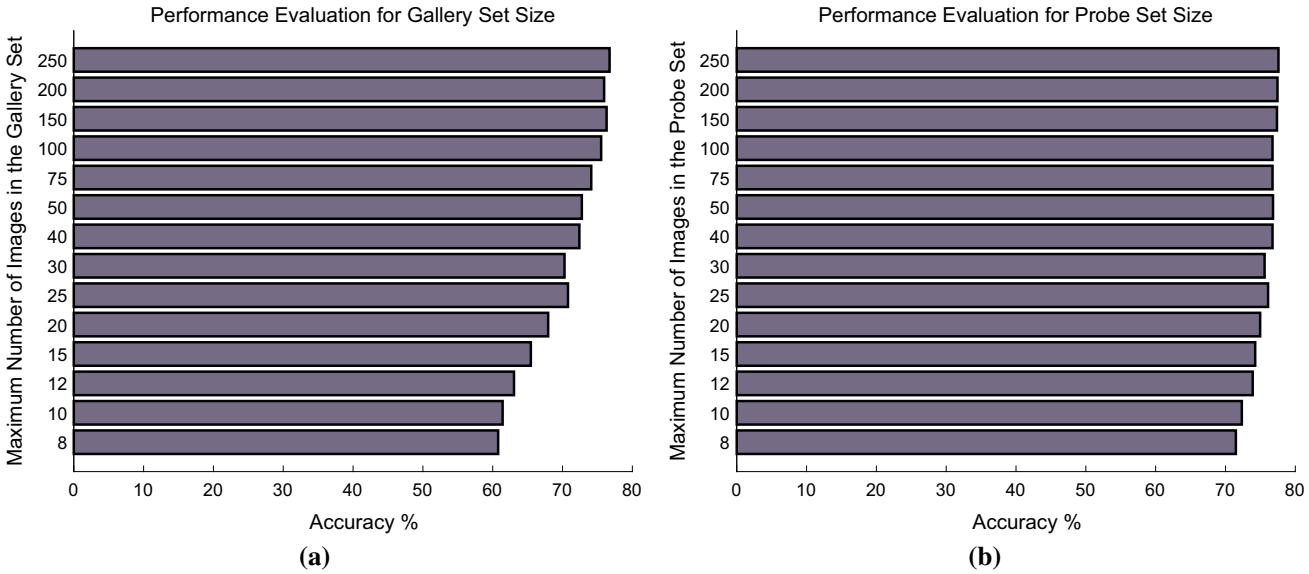

Fig. 8 Classification performance on YouTube celebrities dataset for reduced number of images in the **a** gallery and **b** probe sets

mance drop is the imbalance between $\mathcal{X}_q$ and $\mathcal{D}_1$ for larger values of m. We also perform experiments by excluding the miss-labelled images from $\mathcal{D}_1$. A classification accuracy of 78.8 is achieved for $m = 0$. These evaluations suggests robustness of the proposed method against number of images in $\mathcal{D}_1$ and $\mathcal{D}_2$. Although increasing the size of $\mathcal{D}_1$ increases the number of miss-labelled images, the overall proportion of these images stays the same *i. e.* $\frac{1}{k}$ of all the images. For a large value of k (the number of enrolled subjects in the gallery), the proportion of these images is too small to significantly impact the performance of the proposed method.

(iii) Number of Enrolled Subjects In our experimental evaluations (Sect. 4), the efficacy of the proposed method has been demonstrated on a wide range of datasets in which number of enrolled subjects vary from 20 to 1000. Furthermore, in the previous experiment (Sect. 4.5ii), it was shown that for a larger value of k, the fraction of miss-labelled images in $\mathcal{D}_1$ ($\frac{1}{k}$ of all images) is small and does not significantly affect the training of the binary classifier and the overall performance of the proposed method. In this experiment, we want to quantitatively evaluate the affect of k (the number of enrolled subjects in the gallery) on the performance of the proposed method. In this regards, we perform experiments on YouTube celebrities dataset for $k = \{47, 40, 30, 20, 15\}$. For each value of k, we further do evaluations by considering different number of images in the gallery sets (full length, 200, 100, 50, 25). The experimental results presented in Fig. 9 suggest a gradual performance drop for a reduced number of enrolled subjects in the gallery. The performance drop how-

Table 6 Performance evaluation by changing the number of images in $\mathcal{D}_1$ and $\mathcal{D}_2$

Images	Performance	Images	Performance
0.5 $N_{\mathcal{D}_{1c}}$	77.7 ± 3.8	3 $N_{\mathcal{D}_{1c}}$	74.8 ± 3.5
1 $N_{\mathcal{D}_{1c}}$	77.4 ± 3.5	4 $N_{\mathcal{D}_{1c}}$	73.8 ± 3.3
1.5 $N_{\mathcal{D}_{1c}}$	76.9 ± 3.8	5 $N_{\mathcal{D}_{1c}}$	73.5 ± 3.3
2 $N_{\mathcal{D}_{1c}}$	76.3 ± 3.5	6 $N_{\mathcal{D}_{1c}}$	73.3 ± 3.3
2.5$N_{\mathcal{D}_{1c}}$	75.7 ± 3.5	7 $N_{\mathcal{D}_{1c}}$	73.2 ± 3.3

Originally, $N_{\mathcal{D}_{1c}} = \left\lceil \frac{N_q}{k} \right\rceil$ images are sampled from each class of the training data to from $\mathcal{D}_1$. Here, we evaluate the method by changing the number of these images from $N_{\mathcal{D}_{1c}}$ to $mN_{\mathcal{D}_{1c}}$ where $m = \{0.5, 1, 1.5, 2, 2.5, 3, 4, 5, 6, 7\}$

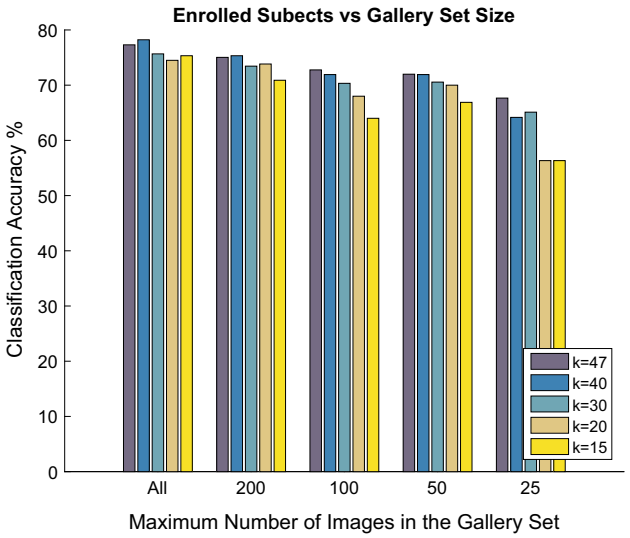

Fig. 9 Performance evaluation for different number of enrolled subjects and maximum number of images in the gallery

ever is quite insignificant when the gallery sets contain more images. The performance drop for lower values of k is more pronounced when the gallery sets contain fewer images.

4.6 Ablative Analysis

We conduct experiments on YouTube celebrities dataset to study the contribution of the different components of the proposed method towards its overall performance. The following aspects are explored:

(i) Binary Classifiers Experiments are performed by considering different binary classifiers which include linear SVM (Fan et al. 2008), non-linear SVM with Radial Basis Function (RBF) kernel (Chang and Lin 2011), non-linear SVM with Chi-Square kernel (Vedaldi and Zisserman 2012) and random decision forests (Breiman 2001). Experimental results in Table 7 show that the choice of the binary classifier does not significantly impact the performance. Although, for many classification tasks, non-linear SVMs perform better compared with linear SVMs, in our case, they show a comparable performance. This can be due to strong discriminative feature representation in terms of activations of a Convolution Neural Network. CNN based features in combination with a linear SVM have shown superior performance for many challenging classification tasks (Sharif Razavian et al. 2014; Khan et al. 2016). We, therefore, select linear SVM because of its computational efficiency. We note that for linearly inseparable data, linear SVM may perform poorly. In such a case, any non-linear binary classifier can easily be employed in conjunction with the proposed technique.

(ii) Feature Descriptors Experiments are performed on YouTube celebrities dataset by considering different methods of encoding facial images. These include Local Binary Patterns (LBPs) (Ojala et al. 2002), Gabor features (Yang et al. 2004), activations of AlexNet (Krizhevsky et al. 2012) fine-tuned on BU4DFE dataset (Yin et al. 2008) and activations of VGG-Face CNN model (Parkhi et al. 2015). For LBPs, each image is divided into 4×4 non-overlapping blocks and 59 dimensional histograms are extracted from each block. Histograms from all 16 blocks are then concatenated to get the final 944 dimensional feature vector. For Gabor features, we generate a bank of 40 Gabor wavelet filters at five scales and eight orientations. An image is then convolved with these filters, and the down sampled magnitude responses are considered as feature representation. For CNN models, we consider the 4096 dimensional activations of the first fully connected layer of the model as feature representation of the input image. Experimental results in Table 8 show that the learned feature representations in terms of activations of CNN models perform significantly better compared with LBPs and Gabor features. We also evaluate these features in combination with Principal Component Analysis (PCA) whitening. The results show that PCA whitening achieves a performance improvement for LBPs and Gabor features, while a slight performance drop for learned features.

(iii) Number of Iterations Fig 10 shows performance evaluation for different number of maximum iterations of steps 1-5 of the proposed method. The results show that performing more iterations improves the robustness of our approach and results in a slightly improved recognition performance. This, however, requires more computational effort. A total of five

Table 7 Performance evaluation for different choices of binary classifiers

Classifier	Accuracy
Linear SVM C_1	74.2 ± 3.6
Linear SVM C_2	73.3 ± 3.7
Non-linear SVM RBF Kernel	74.1 ± 3.5
Non-linear SVM Chi-square Kernel	74.7 ± 3.5
Random decision forests	73.8 ± 3.6

Table 8 Performance evaluation for different feature descriptors

Features	Dimensions	Accuracy
LBPs	944	71.5 ± 3.8
LBPs (PCA Whitening)	400	74.6 ± 3.5
Gabor	4000	70.8 ± 3.7
Gabor (PCA Whitening)	400	73.8 ± 3.8
AlexNet	4096	78.5 ± 3.7
AlexNet (PCA Whitening)	400	77.4 ± 3.5
VGG-Face	4096	86.0 ± 3.4
VGG-Face (PCA Whitening)	400	84.3 ± 3.4

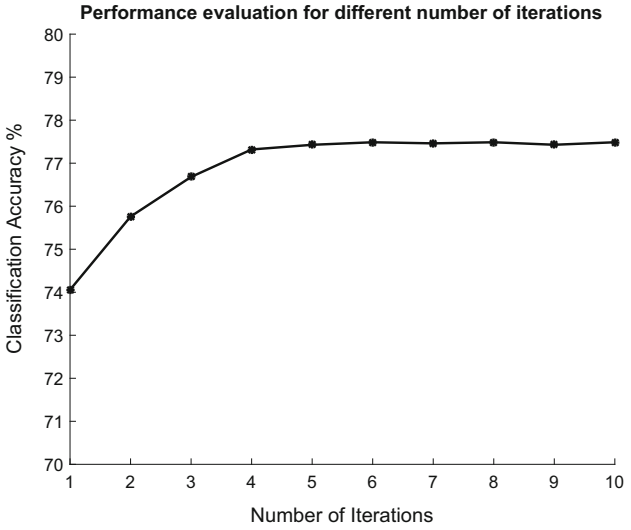

Fig. 10 Performance evaluation for different number of iterations of steps 2–8 (Algorithm 1) of the proposed method. Considering the performance and required computational load, we select a total of five iterations as an optimal choice

Table 9 Ablative analysis for different sampling strategies and ensemble of classifiers

Sampling strategies		Ensemble effect	
Uniform random	74.6	C_1	74.2
Bootstrapped	75.2	C_2	73.3
Posebased	77.4	C_1 and C_2	77.4

iterations is therefore a good trade off between recognition performance and computational complexity.

(iv) Sampling Strategies The results in Table 9 show that bootstrapped sampling introduces more robustness and enhances performance. Incorporating pose based information during sampling further enhances the performance of the proposed method (since most of the images in the sampled set have the same pose as the pose of the images of the query image set). By doing so, the trained binary classifier learns to discriminate between the images of the query set from the others (rather than discriminating them based upon their poses). A visual inspection of the failure cases revealed that most of the miss-classifications happened when the pose difference between most of the images of the gallery and probe set is greater than 45°.

(v) Ensemble Effect The results in Table 9 show that use of the two binary classifiers (see Sec 3.3) complement each other and result in a performance boost.

Based on our empirical evaluations and ablative analysis on YTC dataset, we attribute the performance achieved by our proposed method to the following reasons. The proposed method can naturally accommodate fusion of information from multiple classifiers. This can be a binary classifier trained multiple times for different random samplings of the negative set. Further, it can simultaneously fuse information from different types of binary classifiers. Convolution Neural Networks based learnt feature representations also achieve a significant performance boost for the proposed method.

Table 10 Timing comparison on the YouTube celebrities dataset

Method	Train	Test	Method	Train	Test
MSM	N/A	1.1	SANP	N/A	22.4
DCC	27.9	0.2	CDL	549.6	7.2
MMD	N/A	68.1	MSSRC	N/A	54.2
MDA	7.2	0.1	SSDML	389.3	18.5
AHISD	N/A	3.1	RNP	N/A	1.4
CHISD	N/A	5.3	Ours	N/A	6.5

Time in seconds required for offline training and online testing of one image set on YouTube celebrities dataset. 'N/A' means that the method does not perform any offline training

4.7 Timing Analysis

Table 10 lists the times (in seconds) for different methods using the respective Matlab implementations on a core i7 machine. Specifically, the time required for the offline training and the time needed to test one image set on the YouTube celebrities dataset are provided. The reported time for our method corresponds to five iterations of steps 1–5 of our algorithm (see Sect. 3.2). For MSM (Yamaguchi et al. 1998), AHISD (Cevikalp and Triggs 2010), CHISD (Cevikalp and Triggs 2010) and RNP (Yang et al. 2013), the reported test time also includes the time required to compute subspaces and projection matrices of the training data. These can be computed offline. It takes approximately 0.9 s to compute them for the training data of YouTube celebrities dataset.

Based upon their computational requirements, we can categorize the evaluated methods as online (which do all computations at run time e.g., Yamaguchi et al. 1998; Cevikalp and Triggs 2010; Hu et al. 2012; Ortiz et al. 2013; Yang et al. 2013) and offline (which do training component offline and only testing is done at run time e.g., Kim et al. 2007; Wang and Chen 2009; Wang et al. 2012; Zhu et al. 2013; Yang et al. 2013). Both of these categories of methods have their strengths and limitations. A major strength of online methods is their scalability. New classes can easily be added without requiring retraining on the complete dataset. A major limitation of online methods (including ours) is that all the computation is done at run-time and comparatively more memory storage is required. In our implementation, we noted that, on average, our method requires approximately 450 MB of RAM to classify a query image set on YouTube celebrities dataset. In comparison, offline methods are efficient at run time and require less computational resources.

5 Conclusion

A new approach is introduced to efficiently extend well known binary classifiers for multi-class image set classification. Compared with the popular one-vs-one and one-vs-rest binary to multi-class strategies, the proposed approach is very efficient as it trains fixed number of binary classifiers (one to five) and uses very few images for training. The proposed approach can also simultaneously fuse information from different types of binary classifiers, which further enhances its robustness and accuracy. Extensive experiments have been performed to validate the proposed approach for the tasks of video based face recognition, still to video face recognition and object recognition. The experimental results and a comparison with the existing methods show that the proposed method consistently achieves state of the art performance.

References

An, S., Hayat, M., Khan, S. H., Bennamoun, M., Boussaid, F., & Sohel, F. (2015). Contractive rectifier networks for nonlinear maximum margin classification. In *Proceedings of the IEEE international conference on computer vision* (pp. 2515–2523)

Arandjelovic, O., Shakhnarovich, G., Fisher, J., Cipolla, R., & Darrell, T. (2005). Face recognition with image sets using manifold density divergence. In *2005 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 581–588)

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Cevikalp, H., & Triggs, B. (2010). Face recognition based on image sets. In *IEEE conference on computer vision and pattern recognition, 2010. CVPR 2010* (pp. 2567–2573). IEEE.

Chang, C. C., & Lin, C. J. (2011). Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*(3), 27.

Chatfield, K., Simonyan, K., Vedaldi, A. & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. In BMVC.

Chien, J. T., & Wu, C. C. (2002). Discriminant waveletfaces and nearest feature classifiers for face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(12), 1644–1649.

Davis, J. V., Kulis, B., Jain, P., Sra, S. & Dhillon, I. S. (2007). Information-theoretic metric learning. In *Proceedings of the 24th international conference on machine learning* (pp. 209–216). ACM.

Eth80. http://www.d2.mpi-inf.mpg.de/Datasets/ETH80. Accessed 05 July 2014.

Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., & Lin, C. J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, *9*, 1871–1874.

Fanelli, G., Gall, J., & Van Gool, L. (2011a). Real time head pose estimation with random regression forests. In *2011 IEEE conference on computer vision and pattern recognition (CVPR)* pp. 617–624. IEEE.

Fanelli, G., Weise, T., Gall, J., & Van Gool, L. (2011b). Real time head pose estimation from consumer depth cameras. *Pattern Recognition*, *6835*, 101–110.

Goldberger, J., Roweis, S., Hinton, G., & Salakhutdinov, R. (2004). Neighbourhood components analysis. In *Advances in neural information processing systems*, (p. 17).

Gross, R., & Shi, J. (2001). The cmu motion of body (mobo) database. Technical report.

Harandi, M. T., Sanderson, C., Shirazi, S., & Lovell, B. C. (2011). Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching. In *2011 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2705–2712).

Hayat, M., & Bennamoun, M. (2014). An automatic framework for textured 3d video-based facial expression recognition. *IEEE Transactions on Affective Computing*, *5*(3), 301–313.

Hayat, M., Bennamoun, M. & An, S. (2014). Learning non-linear reconstruction models for image set classification. In *2014 IEEE conference on computer vision and pattern recognition (CVPR)*.

Hayat, M., Bennamoun, M. & An, S. (2014). Reverse training: An efficient approach for image set classification. In: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (eds.) Computer Vision ECCV 2014, *Lecture Notes in Computer Science*, vol. 8694, pp. 784–799. Springer International Publishing.

Hayat, M., Bennamoun, M., & An, S. (2015). Deep reconstruction models for image set classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(4), 713–727.

Hayat, M., Bennamoun, M. & El-Sallam, A. A. (2013). Clustering of video-patches on grassmannian manifold for facial expression recognition from 3d videos. In *2013 IEEE workshop on applications of computer vision (WACV)*.

Hu, Y., Mian, A. S., & Owens, R. (2012). Face recognition using sparse approximated nearest points between image sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(10), 1992–2004.

Huang, Z., Shan, S., Zhang, H., Lao, S., Kuerban, A. & Chen, X. (2013). Benchmarking still-to-video face recognition via partial and local linear discriminant analysis on COX-S2V dataset. In *Computer Vision–ACCV 2012* (pp. 589–600). Springer.

Huang, Z., Wang, R., Shan, S. & Chen, X. (2014). Learning euclidean-to-riemannian metric for point-to-set classification.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S. & Darrell, T. (2014), Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093.

Khan, S. H., Bennamoun, M., Sohel, F. & Togneri, R. (2014). Automatic feature learning for robust shadow detection. In *IEEE 27th international conference on computer vision and pattern recognition (CVPR)* (pp. 1939–1946). IEEE.

Khan, S. H., Hayat, M., Bennamoun, M., Togneri, R., & Sohel, F. A. (2016). A discriminative representation of convolutional features for indoor scene recognition. *IEEE Transactions on Image Processing*, *25*(7), 3372–3383.

Kim, M., Kumar, S., Pavlovic, V. & Rowley, H. (2008). Face tracking and recognition with visual constraints in real-world videos. In *2008 IEEE conference on computer vision and pattern recognition (CVPR)*, (pp. 1–8). IEEE.

Kim, T. K., Kittler, J., & Cipolla, R. (2007). Discriminative learning and recognition of image set classes using canonical correlations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(6), 1005–1018.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS* (pp. 1097–1105).

Kumar, N., Berg, A.C., Belhumeur, P. N., & Nayar, S. K. (2009). Attribute and simile classifiers for face verification. In *IEEE international conference on computer vision (ICCV)*.

Lee, K. C., Ho, J., Yang, M. H. & Kriegman, D. (2003). Video-based face recognition using probabilistic appearance manifolds. In *2003 IEEE conference on computer vision and pattern recognition (CVPR)*, vol. 1, pp. I–313. IEEE.

Leibe, B. & Schiele, B. (2003). Analyzing appearance and contour based methods for object categorization. In *2003 IEEE conference on*

computer vision and pattern recognition (CVPR) vol. 2, pp. II–409. IEEE.

Li, B. Y., Mian, A. S., Liu, W. & Krishna, A. (2013). Using kinect for face recognition under varying poses, expressions, illumination and disguise. In *2013 IEEE workshop on applications of computer vision (WACV)* (pp. 186–192). IEEE.

Lu, J., Wang, G. & Moulin, P. (2013). Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning. In *2013 IEEE conference on international conference on computer vision (ICCV)*

Ng, H. W. & Winkler, S. (2014). A data-driven approach to cleaning large face datasets. In *IEEE international conference on image processing, Paris, France*, 27–30 Oct. IEEE.

Oja, E. (1983). *Subspace methods of pattern recognition* (Vol. 4). Baldock: Research Studies Press England.

Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(7), 971–987.

Ortiz, E., Wright, A. & Shah, M. (2013). Face recognition in movie trailers via mean sequence sparse representation-based classification. In *2013 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3531–3538). doi:10.1109/CVPR.2013.453

Parkhi, O. M., Vedaldi, A. & Zisserman, A.(2015). Deep face recognition. In *British machine vision conference*.

Ross, D. A., Lim, J., Lin, R. S., & Yang, M. H. (2008). Incremental learning for robust visual tracking. *International Journal of Computer Vision*, *77*(1–3), 125–141.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, *115*(3), 211–252. doi:10.1007/s11263-015-0816-y.

Shakhnarovich, G., Fisher, J. W., & Darrell, T. (2002). Face recognition from long-term observations. In *European conference on computer vision (ECCV)*, (pp. 851–865). Springer.

Sharif Razavian, A., Azizpour, H., Sullivan, J. & Carlsson, S. (2014). Cnn features off-the-shelf: An astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 806–813).

Sharma, A., Kumar, A., Daume, H. & Jacobs, D. W. (2012). Generalized multiview analysis: A discriminative latent space. In *2012 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2160–2167). IEEE.

Sugiyama, M. (2007). Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. *The Journal of Machine Learning Research*, *8*, 1027–1061.

Uzair, M., Mahmood, A., Mian, A. & McDonald, C. (2013). A compact discriminative representation for efficient image-set classification with application to biometric recognition. In *2013 International conference on biometrics (ICB)*. IEEE.

Vedaldi, A., & Zisserman, A. (2012). Efficient additive kernels via explicit feature maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(3), 480–492.

Vincent, P. & Bengio, Y. (2001). K-local hyperplane and convex distance nearest neighbor algorithms. In *Advances in neural information processing systems* (pp. 985–992).

Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, *57*(2), 137–154.

Wang, R. & Chen, X. (2009). Manifold discriminant analysis. In *IEEE conference on computer vision and pattern recognition, 2009. CVPR 2009*, (pp. 429–436). IEEE.

Wang, R., Guo, H., Davis, L. S. & Dai, Q. (2012). Covariance discriminative learning: A natural and efficient approach to image set classification. In *2012 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2496–2503). IEEE.

Wang, R., Shan, S., Chen, X. & Gao, W. (2008). Manifold-manifold distance with application to face recognition based on image set. In *IEEE conference on computer vision and pattern recognition, 2008. CVPR 2008* (pp. 1–8). IEEE.

Weinberger, K. Q., & Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, *10*, 207–244.

Yamaguchi, O., Fukui, K. & Maeda, K. I. (1998). Face recognition using temporal image sequence. In *1998 IEEE international conference on automatic face and gesture recognition (FG)* (pp. 318–323). IEEE.

Yang, M., Zhu, P., Gool, L. V. & Zhang, L. (2013). Face recognition based on regularized nearest points between image sets, pp. 1–7.

Yang, P., Shan, S., Gao, W., Li, S. Z. & Zhang, D. (2004). Face recognition using ada-boosted gabor features. In*Proceedings on sixth IEEE international conference on automatic face and gesture recognition, 2004* (pp. 356–361). IEEE.

Yin, L., Chen, X., Sun, Y., Worm, T. & Reale, M. (2008). A high-resolution 3d dynamic facial expression database. In *8th IEEE international conference on automatic face gesture recognition, FG '08* (pp. 1 –6).

Zhu, P., Zhang, L., Zuo, W. & Zhang, D. (2013). From point to set: Extend the learning of distance metrics. In *2013 IEEE conference on international conference on computer vision (ICCV)*. IEEE.