# Reverse Training: An efficient Approach for Image Set Classification

Munawar Hayat, Mohammed Bennamoun, Senjian An

School of Computer Science and Software Enginnering
The University of Western Australia

**Abstract.** This paper introduces a new approach, called *reverse training*, to efficiently extend binary classifiers for the task of multi-class image set classification. Unlike existing binary to multi-class extension strategies, which require multiple binary classifiers, the proposed approach is very efficient since it trains a single binary classifier to optimally discriminate the class of the query image set from all others. For this purpose, the classifier is trained with the images of the query set (labelled positive) and a randomly sampled subset of the training data (labelled negative). The trained classifier is then evaluated on rest of the training images. The class of these images with their largest percentage classified as positive is predicted as the class of the query image set. The confidence level of the prediction is also computed and integrated into the proposed approach to further enhance its robustness and accuracy. Extensive experiments and comparisons with existing methods show that the proposed approach achieves state of the art performance for face and object recognition on a number of datasets.

**Keywords:** Image Set Classification, Face and Object Recognition

## 1 Introduction

Face or object recognition is traditionally treated as a single image based classification problem, that is, given a single query image, we are required to find its best match in a gallery of images. However, in many real-world applications (e.g. recognition from surveillance videos, multi-view camera networks and personal albums), multiple images of a person or an object are readily available. Recognition from these multiple images is studied under the framework of image set classification. Classification from image sets (as opposed to single image based classification) is more promising as it aims to effectively handle a wide range of appearance variations, which are commonly present within images of the same object in an image set. These variations can be caused by changing lighting conditions, different view points, non-rigid deformations and occlusions [5, 12, 17]. For these reasons, image set classification has attained significant research attention in recent years [3, 8, 9, 11, 14, 20, 25–27, 29, 30].

Although image set classification provides a plentitude of data of the same object under different variations, it simultaneously poses many challenges to

make effective use of this data. The major focus of the existing image set classification methods has therefore been to find a suitable representation which can effectively model the appearance variations within images of an image set. For example, the methods in [10,14,19,25,27,28] use subspaces to model image sets, and set representative exemplars (generated from affine hull/convex hull) are used in [3,11] for image set representations. The mean of the set images is used for set representation in [11, 18, 20] and image sets are represented as a point on a manifold geometry in [8, 26]. 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 information contained in the images of the image set. In this paper, we take a different approach and avoid representing an image set by a single entity. We retain the images of the image set in their original form and instead design an efficient classification framework to effectively deal with the plentitude 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, they can be adapted for image set classification by first individually classifying images of a query set and then devising an appropriate voting strategy (see Sec 4.2). However, due to the plentitude of data involved in the case of image set classification, a straight forward extension of these algorithms from single image based to image set classification would be computationally burdensome. Specifically, since most of the popular learning algorithms (e.g. Support Vector Machines, AdaBoost, 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 training of multiple binary classifiers. One-vs-one and one-vs-rest are the two most commonly adapted 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. Although, one-vs-rest trains comparatively fewer classifiers, it requires images from all classes to train each binary classifier. Adapting 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 has to be trained.

The framework proposed in this paper trains a very few 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 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 equal to the number of images of the query image set. 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 other

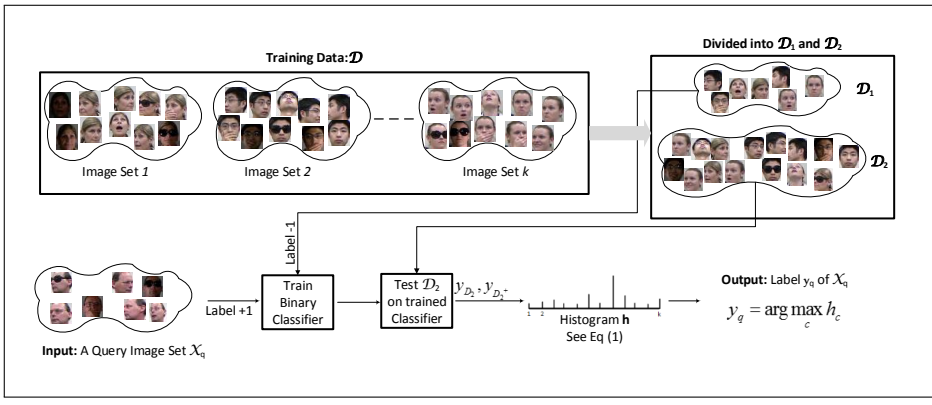


Fig. 1: Block diagram of the proposed method. Training data is divided into two sets $\mathcal{D}_1$ and $\mathcal{D}_2$. $\mathcal{D}_1$ contains uniformly randomly sampled images from all classes such that the size of $\mathcal{D}_1$ is the same as the size of the query image set $\mathcal{X}_q$. A binary classifier is trained, with images of $\mathcal{X}_q$ (labeled $+1$) and $\mathcal{D}_1$ (labeled $-1$). 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 2 for illustration.

classes. Next, the learned classifier is evaluated on the images of $\mathcal{D}_2$ ($\mathcal{D}_2$ contains all training images except the ones in $\mathcal{D}_1$). The images of $\mathcal{D}_2$ which are classified to belong to the images of the query set are of our interest. Knowing the original class labels of these training images, we formulate a histogram which is then used to decide about the class of the query set. A complete detailed description of the proposed framework is presented in Sec 3 along with an illustration using a toy example in Fig. 2.

The main strengths and contributions of this paper are as follows. 1) A new concept 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, one-vs-rest), the proposed approach is computationally very efficient. It only requires training of a fixed number of binary classifiers (1 to 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 potential miss-classification. Knowing pre-hand about a query image set being miss-classified makes it possible to use 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 with the addition of new classes in the gallery (see

Sec. 4.2). Many 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 query class from other classes using a fixed number of binary classifiers.

## 2 Related Work

The major challenge addressed by the existing research on image set classification has been to find a representation which can effectively model the appearance variations within images of an image set. Two different approaches have been adopted for this purpose. The **first** approach models the variations within images of a set by a statistical distribution and uses a measure such as KL-divergence to compare two sets. Methods based on this approach are called parametric model based methods [2,22]. One major limitation of these methods 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 [3,8,9,11,14,20,23,25–27,29,30] 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 [14, 28], orthogonal subspaces [19] and a combination of linear subspaces [25,27]. 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 dimensions. 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) [3] and Sparse Approximated Nearest Points (SANP) [11] use affine hull, whereas Convex Hull Image Set Distance (CHISD) [3] uses the convex hull of the images to model an image set. The 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-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 [8, 25] and Lie group of Riemannian manifold [26]. The mean of the set images is also used for image set representation in [11, 18, 20].

In this paper, we argue that a single entity (e.g. a mean image, a subspace, a point on a manifold, an exemplar generated from a geometric structure) for

image set representation can be sub-optimal, insufficient 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. In the case of image sets represented by their mean images, the mean image could be visually very different from the rest of the images in the set. For illustration purposes, consider taking the mean of two face images from the right and left profile views. The mean image would be blurred and contain two superimposed faces. 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 an altogether different approach which does not require any 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 Framework

**Problem Description:** For $k$ classes of a training data, we are given $k$ image sets $\mathcal{X}_1, \mathcal{X}_2, \cdots \mathcal{X}_k$ and their corresponding class labels $y_c \in [1, 2, \cdots k]$. An image set $\mathcal{X}_c = \{\mathbf{x}^{(t)} | y^{(t)} = c; t = 1, 2, \cdots N_c\}$ 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 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 $\mathcal{X}_q$.

### 3.1 Image Set Classification Algorithm

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

1. Images from all training sets are gathered into a single set $\mathcal{D} = \{\mathcal{X}_1, \mathcal{X}_2, \cdots \mathcal{X}_k\}$. Next, $\mathcal{D}$ is divided into two sets: $\mathcal{D}_1$ and $\mathcal{D}_2$. Let $\mathcal{D}_{1c}$ be a randomly sampled subset of $\mathcal{X}_c$ with a set size $N_{\mathcal{D}_{1c}}$, where $N_{\mathcal{D}_{1c}} = \frac{N_q}{k}$ rounded to the nearest integer, then the set $\mathcal{D}_1$ is formed by the union $\bigcup_c \mathcal{D}_{1c}$, $c = 1, 2, \cdots k$. $\mathcal{D}_2$ is achieved by $\mathcal{D}_2 = \mathcal{D} \setminus \mathcal{D}_1$. The class label information of images in $\mathcal{D}_1$ and $\mathcal{D}_2$ is stored in sets $\mathbf{y}_{\mathcal{D}_1} = \{y^{(t)} \in [1, 2, \cdots k], t = 1, 2, \cdots N_{\mathcal{D}_1}\}$ and $\mathbf{y}_{\mathcal{D}_2} = \{y^{(t)} \in [1, 2, \cdots k], t = 1, 2, \cdots N_{\mathcal{D}_2}\}$ 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 labeled $+1$ while the images in $\mathcal{D}_1$ are labeled $-1$. 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 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 selected binary classifier (see Sec 3.2) treats

them as outliers and learns to discriminate the class of the query image set from all other classes.

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 $c$th 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,

$$\mathbf{h}_c = \frac{\sum\limits_{y^{(t)} \in \mathbf{y}_{\mathcal{D}_2^+}} \delta_c(y^{(t)})}{\sum\limits_{y^{(t)} \in \mathbf{y}_{\mathcal{D}_2}} \delta_c(y^{(t)})} \text{ ,where}$$
$$\delta_c(y^{(t)}) = \begin{cases} 1, & y^{(t)} = c \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

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 the 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 the confidence $d$ is greater than a certain threshold. The value of the threshold is determined empirically by performing experiments on a cross 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 if a maximum of five iterations have already been done. Doing more iterations enhances the robustness of the method (since different images

**Algorithm 1** The proposed Image Set Classification algorithm

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

$\mathcal{D} \leftarrow \{\mathcal{X}_1, \mathcal{X}_2, \cdots \mathcal{X}_k\}$ ▷ $\mathcal{D}$: All training images
$\mathcal{D}_1 \leftarrow \bigcup_c \mathcal{D}_{1c}$ where $\mathcal{D}_{1c}$ is a random subset of $\mathcal{X}_c$
$\mathcal{D}_2 \leftarrow \mathcal{D} \setminus \mathcal{D}_1$ ▷ $\mathcal{D}$ 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 $-1$
$l_{\mathcal{D}_2} \leftarrow test(C_1, \mathcal{D}_2)$ ▷ Test $\mathcal{D}_2$ on classifier $C_1$
$y_{\mathcal{D}_2^+} \leftarrow l_{\mathcal{D}_2}, y_{\mathcal{D}_2}$ ▷ labels of images of $\mathcal{D}_2$ 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** $d > th$ **then**
        $y_q \leftarrow \arg\max_c \bar{\mathbf{h}}_c$
    **else**
        $y_q \leftarrow$ Repeat for another binary classifier $C_2$
    **end if**
**end if**

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

are selected into $\mathcal{D}_1$ and $\mathcal{D}_2$ for every iteration) but at the cost of increased computational effort. Our experiments revealed that a maximum of five iterations is a good trade-off between the robustness and the computational efficiency.

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 miss-classification. We therefore seek the opinion of 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 about the choice of the binary classifiers $C_1$ and $C_2$ is given next.

### 3.2 The Choice of the 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 such that it should generalize well to unseen data while testing. Moreover, since the binary classifier is being trained on images of $\mathcal{X}_q$ and $\mathcal{D}_1$ and some images in $\mathcal{D}_1$ have the same class as of $\mathcal{X}_q$, the binary classifier should treat these images as outliers. For these reasons, Support Vector Machine (SVM) with a linear Kernel is

deemed to be an appropriate choice. It is known to show 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 the linear SVM with **L2** regularization and **L2** loss function, while $C_2$ is the linear SVM with **L1** regularization and **L2** loss function [4]. 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,

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

while, $C_2$ solves the following optimization problem,

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

Here $\mathbf{w}$ is the coefficient vector to be learned and $C > 0$ is the penalty parameter used for regularization. After learning 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 the classifier $C_2$ (trained for challenging examples) is sparse. Learning a sparse $\mathbf{w}$ for $C_2$ further enhances the generalization for the challenging cases.

### 3.3 Illustration with a toy Example

The proposed image set classification algorithm is illustrated with the help of a toy example in Fig 2. 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 2 (a). First, we form $\mathcal{D}_1$ by randomly sampling points from $\mathcal{X}_1$, $\mathcal{X}_2$ and $\mathcal{X}_3$. Fig 2 (b) shows the datapoints of $\mathcal{D}_1$ and $\mathcal{X}_q$. Next, a linear SVM is trained by labeling the datapoints of $\mathcal{X}_q$ as $+1$ and $\mathcal{D}_1$ as $-1$. Note that SVM (Fig 2 (c)) ignores the miss-labeled 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. Fig 2 (d) 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 evaluate the performance of the proposed method for the task of image set classification with applications to face and object recognition. For face recognition, we perform experiments on three video datasets (Honda/UCSD [15], CMU Mobo [7], YouTube Celebrities [13]) and an RGB-D Kinect dataset (obtained by combining three Kinect datasets). For object recognition, we use ETH-80 dataset [16]. Below, we first give a brief description of each of these datasets followed by the adopted experimental configurations. We then present a performance comparison of the proposed method with the baseline multi-class classification strategies (Sec. 4.2). Finally, in Sec. 4.3, we compare our method with the existing state of the art image set classification methods.

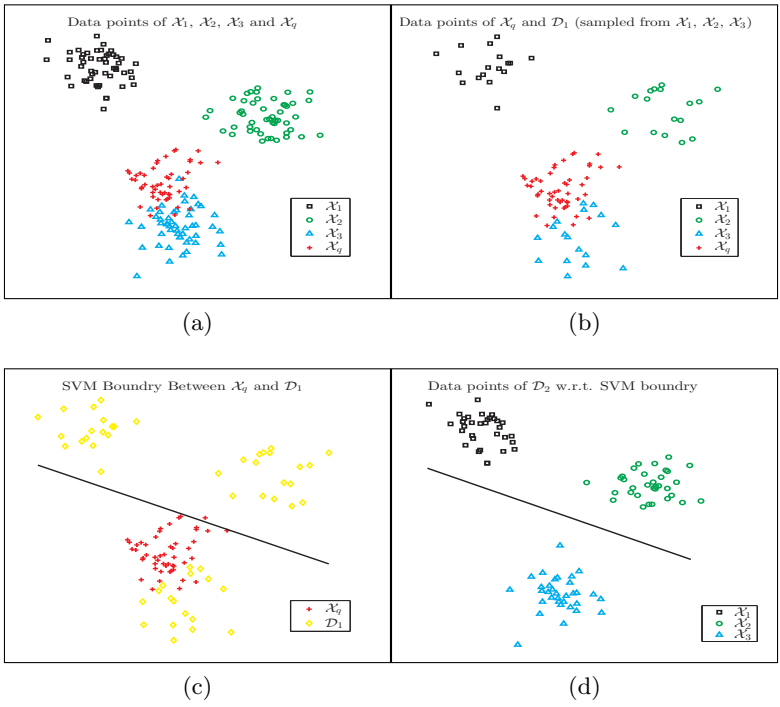


Fig. 2: Toy example to illustrate the proposed method. Consider a training data with three classes and the task is to find the class of $\mathcal{X}_q$ **(a)**. Data points from three training image sets $\mathcal{X}_1$, $\mathcal{X}_2$, $\mathcal{X}_3$ and a query image set $\mathcal{X}_q$ are shown. **(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 $\mathcal{X}_q$ (labeled $+1$) and $\mathcal{D}_1$ (labeled $-1$). **(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 boundry as the points of $\mathcal{X}_q$, the proposed method declares $\mathcal{X}_q$ to be from $\mathcal{X}_3$. Figure best seen in colour.

### 4.1 Evaluated Datasets and Experimental Settings

**The Honda/UCSD dataset [15]** contains 59 video sequences (with 12 to 645 frames in each video) of 20 subjects. We use Viola and Jones face detection [24] algorithm to extract faces from video frames. The extracted faces are then resized to $20 \times 20$. For our experiments, we consider each video sequence as an image set and follow the standard evaluation configuration provided in [15]. 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 training and testing image sets.

**The CMU Mobo (Motion of Body) dataset [7]** contains a total of 96 video sequences of 24 subjects walking on a treadmill. The faces from the videos are extracted using [24] and resized to $40 \times 40$. Similar to [11, 26], we consider each video as an image set and use one set per subject for training and the remaining sets for testing. To achieve a consistency, experiments are repeated ten times for different training and testing sets.

**YouTube Celebrities [13]** 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 therefore exhibit 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 [24] fails for a significant number of frames for this dataset. We therefore use tracking [21] 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.* [21] to track the face region in the subsequent frames. The extracted face region is then resized to $30 \times 30$. In order to perform experiments, we treat the faces acquired from each video as an image set and follow the five fold cross validation experimental setup similar to [11,25–27]. 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 [17], Biwi Kinect [5] 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 forrest regression based classifier of [6] 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, sunglass 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 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 $128 \times 128$ cropped images [1] and resize them to $32 \times 32$. We follow an experimental setup similar to [14, 25, 26]. 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. 10 runs of experiments are performed for different random selections of the training and testing sets.

Table 1: Performance Comparison with the baseline methods

| Methods | Honda | Mobo | YouTube | Kinect | ETH |
|---|---|---|---|---|---|
| one-vs-one | $92.1 \pm 2.2$ | $94.7 \pm 2.0$ | $67.7 \pm 4.0$ | $94.3 \pm 3.5$ | $96.2 \pm 2.9$ |
| one-vs-rest | $94.6 \pm 1.9$ | $96.7 \pm 1.6$ | $68.4 \pm 4.2$ | $94.6 \pm 3.3$ | $97.6 \pm 1.5$ |
| **This Paper** | $\mathbf{100.0 \pm 0.0}$ | $\mathbf{97.8 \pm 0.7}$ | $\mathbf{74.1 \pm 3.5}$ | $\mathbf{98.1 \pm 1.9}$ | $\mathbf{95.5 \pm 2.0}$ |

Average identification rates of our method and two well-known multi-class classification strategies. The proposed method achieves good performance on all five datasets. 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 and then use 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. The results presented for Honda/UCSD dataset are only for full-lengths of 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.

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 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 classifiers trained 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 a few binary classifier (a maximum of five for the challenging cases) and the number of images used for training is also small.

Table 3: Performance Comparison on Honda/UCSD dataset

| | MSM | DCC | MMD | MDA | AHISD | CHISD |
|---|---|---|---|---|---|---|
| **All** | $88.2 \pm 3.8$ | $92.5 \pm 2.2$ | $92.0 \pm 2.2$ | $94.3 \pm 3.3$ | $91.2 \pm 1.7$ | $93.6 \pm 1.6$ |
| **100** | $85.6 \pm 4.3$ | $89.2 \pm 2.4$ | $85.5 \pm 2.1$ | $91.7 \pm 1.6$ | $90.7 \pm 3.2$ | $91.0 \pm 1.7$ |
| **50** | $83.0 \pm 1.7$ | $82.0 \pm 3.3$ | $83.1 \pm 4.4$ | $85.6 \pm 5.8$ | $89.8 \pm 2.1$ | $90.5 \pm 2.0$ |
| | SANP | CDL | MSSRC | SSDML | RNP | **This Paper** |
| **All** | $95.1 \pm 3.0$ | $98.9 \pm 1.3$ | $97.9 \pm 2.6$ | $86.4 \pm 3.6$ | $95.9 \pm 2.1$ | $\mathbf{100.0 \pm 0.0}$ |
| **100** | $94.1 \pm 3.2$ | $96.2 \pm 1.2$ | $96.9 \pm 1.3$ | $84.3 \pm 2.2$ | $92.3 \pm 3.2$ | $\mathbf{99.7 \pm 0.8}$ |
| **50** | $91.9 \pm 2.7$ | $93.9 \pm 2.2$ | $94.3 \pm 1.4$ | $83.4 \pm 1.7$ | $90.2 \pm 3.2$ | $\mathbf{99.4 \pm 1.1}$ |

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

### 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 [28], Discriminant Canonical Correlation Analysis (DCC) [14], Manifold-to-Manifold Distance (MMD) [27], Manifold Discriminant Analysis (MDA) [25], the Linear version of the Affine Hull-based Image Set Distance (AHISD) [3], the Convex Hull-based Image Set Distance (CHISD) [3], Sparse Approximated Nearest Points (SANP) [11], Covariance Discriminative Learning (CDL) [26], Mean Sequence Sparse Representation Classification (MSSRC) [20], Set to Set Distance Metric Learning (SSDML) [30] and Regularized Nearest Points (RNP) [29]. We use the implementations provided by the respective authors for all methods except CDL. We carefully implemented CDL since it is not publicly available. The parameters for all methods are optimized for best performance.

The experimental results in terms of the average identification rates along with standard deviations of different methods on Honda/UCSD dataset are presented in Table 3. The proposed method achieves perfect classification for all frames of the video sequence considered as an image set. Once the total number of images in the set is reduced to 100 and 50, the average identification rates archived by the method are 99.7% and 99.4% respectively. This suggests the robustness of the method w.r.t. the number of images in the set and its suitability for real-life scenarios with a limited availability of images in the set.

The average identification rates and standard deviations for different methods on CMU/Mobo, YouTube Celebrities, Kinect and ETH datasets are summarized in Table 4. The results suggest that the proposed method outperforms most of

Table 4: Performance on CMU/Mobo, YouTube, Kinect and ETH-80 datasets

| Methods | Mobo | YouTube | Kinect | ETH |
|---|---|---|---|---|
| MSM FG'98 [28] | $96.8 \pm 2.0$ | $50.2 \pm 3.6$ | $89.3 \pm 4.1$ | $75.5 \pm 4.8$ |
| DCC TPAMI'07 [14] | $88.8 \pm 2.4$ | $51.4 \pm 5.0$ | $92.5 \pm 2.0$ | $91.7 \pm 3.7$ |
| MMD CVPR'08 [27] | $92.5 \pm 2.9$ | $54.0 \pm 3.7$ | $93.9 \pm 2.2$ | $77.5 \pm 5.0$ |
| MDA CVPR'09 [25] | $80.9 \pm 12.3$ | $55.1 \pm 4.5$ | $93.4 \pm 3.6$ | $77.2 \pm 5.5$ |
| AHISD CVPR'10 [3] | $92.9 \pm 2.1$ | $61.5 \pm 5.6$ | $91.6 \pm 2.2$ | $78.7 \pm 5.3$ |
| CHISD CVPR'10 [3] | $96.5 \pm 1.2$ | $60.4 \pm 5.9$ | $92.7 \pm 1.9$ | $79.5 \pm 5.3$ |
| SANP TPAMI'12 [11] | $97.6 \pm 0.9$ | $65.6 \pm 5.6$ | $93.8 \pm 3.1$ | $77.7 \pm 7.3$ |
| CDL CVPR'12 [26] | $90.0 \pm 4.4$ | $56.4 \pm 5.3$ | $94.5 \pm 1.0$ | $77.7 \pm 4.2$ |
| RNP FG'13 [29] | $96.1 \pm 1.4$ | $65.8 \pm 5.4$ | $96.2 \pm 2.5$ | $81.0 \pm 3.2$ |
| MSSRC CVPR'13 [20] | $97.5 \pm 0.9$ | $59.4 \pm 5.7$ | $95.5 \pm 2.3$ | $90.5 \pm 3.1$ |
| SSDML ICCV'13 [30] | $95.1 \pm 2.2$ | $66.2 \pm 5.2$ | $86.9 \pm 3.4$ | $81.0 \pm 6.6$ |
| **This Paper** | $\mathbf{97.8 \pm 0.7}$ | $\mathbf{74.1 \pm 3.5}$ | $\mathbf{98.1 \pm 1.9}$ | $\mathbf{95.5 \pm 2.0}$ |

Experimental performance of different methods in terms of average identification rates and standard deviations on CMU/Mobo, YouTube Celebrities, Kinect and ETH-80 datasets. The proposed method achieves the best performance on all four datasets. Especially, the performance improvement is more significant for YouTube and ETH-80 datasets.

the existing methods on all datasets. The difference in the performance is more significant for YouTube Celebrities dataset which is the most challenging dataset since the videos have been acquired in real-life scenarios and the resolution of the face images is very low due to the high compression.

**Timing Comparison:** Table 5 lists the times (in seconds) for different methods using the respective Matlab implementations on a core i7 machine. Specifically, the time required for offline training and the time needed to test one image set on YouTube Celebrities dataset are provided. The reported time for our method is for five iterations of steps 1-5 of our algorithm (see Sec. 3.1). It should be noted that many of the existing methods [3, 11, 20, 28, 29] as well as our method are online. Online methods do not perform any offline training and can easily adapt to newly added and previously unseen training data. However, one major limitation of our method and the existing online methods is that all the computation is done at run-time and comparatively more memory storage is required.

Table 5: Timing Comparison on YouTube Celebrities dataset

| Method | MSM | DCC | MMD | MDA | AHISD | CHISD | SANP | CDL | MSSRC | SSDML | RNP | Ours |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Train | N/A | 27.9 | N/A | 7.2 | N/A | N/A | N/A | 549.6 | N/A | 389.3 | N/A | N/A |
| Test | 1.1 | 0.2 | 68.1 | 0.1 | 3.1 | 5.3 | 22.4 | 7.2 | 54.2 | 18.5 | 0.5 | 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.

### 4.4 Analysis and Discussions

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, classification decisions from multiple classifiers can be fused to enhance the overall performance of the method.

A visual inspection of the challenging YouTube Celebrities dataset 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 the training images of the dataset. For such cases, only those images in $\mathcal{D}_2$ which have the same pose as that of images of $\mathcal{X}_q$ (irrespective of their classes) are classified as $+1$. In our future work, we plan to develop a method to estimate the pose of the face images. The pose information will then be used to sample 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 $\mathcal{D}_2$ will be such that only the images with the right profile views will be considered. This will help to overcome the bias in the classification due to head pose.

## 5 Conclusion

This paper introduced a new concept which is embedded in a framework to extend the 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 a fixed number of binary classifiers (one to five) and uses very few images for training. The proposed method has been evaluated for the task of video based face recognition on Honda/UCSD, CMU/Mobo & YouTube Celebrities datasets, RGB-D face recognition from a Kinect dataset and object recognition from ETH-80 dataset. The experimental results and a comparison with the existing methods show that the proposed method consistently achieves the state of the art performance.

## Acknowledgements

This work is supported by SIRF scholarship from The University of Western Australia (UWA) and Australian Research Council (ARC) grant DP110102166.

## References


1. Eth80. http://www.d2.mpi-inf.mpg.de/Datasets/ETH80, accessed: 2014-07-05
2. Arandjelovic, O., Shakhnarovich, G., Fisher, J., Cipolla, R., Darrell, T.: Face recognition with image sets using manifold density divergence. In: Computer Vision and Pattern Recognition (CVPR), 2005 IEEE Conference on. pp. 581–588. IEEE (2005)
3. Cevikalp, H., Triggs, B.: Face recognition based on image sets. In: Computer Vision and Pattern Recognition, 2010. CVPR 2010. IEEE Conference on. pp. 2567–2573. IEEE (2010)
4. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9, 1871–1874 (2008)
5. Fanelli, G., Gall, J., Van Gool, L.: Real time head pose estimation with random regression forests. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 617–624. IEEE (2011)
6. Fanelli, G., Weise, T., Gall, J., Van Gool, L.: Real time head pose estimation from consumer depth cameras. Pattern Recognition pp. 101–110 (2011)
7. Gross, R., Shi, J.: The cmu motion of body (mobo) database. Tech. rep. (2001)
8. Harandi, M.T., Sanderson, C., Shirazi, S., Lovell, B.C.: Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 2705–2712. IEEE (2011)
9. Hayat, M., Bennamoun, M., An, S.: Learning non-linear reconstruction models for image set classification. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on (2014)
10. Hayat, M., Bennamoun, M., El-Sallam, A.A.: Clustering of video-patches on grassmannian manifold for facial expression recognition from 3d videos. In: Applications of Computer Vision (WACV), 2013 IEEE Workshop on (2013)
11. Hu, Y., Mian, A.S., Owens, R.: Face recognition using sparse approximated nearest points between image sets. Pattern Analysis and Machine Intelligence, IEEE Transactions on 34(10), 1992–2004 (2012)
12. Khan, S.H., Bennamoun, M., Sohel, F., Togneri, R.: Automatic feature learning for robust shadow detection. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on (2014)
13. Kim, M., Kumar, S., Pavlovic, V., Rowley, H.: Face tracking and recognition with visual constraints in real-world videos. In: Computer Vision and Pattern Recognition (CVPR), 2008 IEEE Conference on. pp. 1–8. IEEE (2008)
14. Kim, T.K., Kittler, J., Cipolla, R.: Discriminative learning and recognition of image set classes using canonical correlations. Pattern Analysis and Machine Intelligence, IEEE Transactions on 29(6), 1005–1018 (2007)
15. Lee, K.C., Ho, J., Yang, M.H., Kriegman, D.: Video-based face recognition using probabilistic appearance manifolds. In: Computer Vision and Pattern Recognition (CVPR), 2003 IEEE Conference on. vol. 1, pp. I–313. IEEE (2003)
16. Leibe, B., Schiele, B.: Analyzing appearance and contour based methods for object categorization. In: Computer Vision and Pattern Recognition (CVPR), 2003 IEEE Conference on. vol. 2, pp. II–409. IEEE (2003)
17. Li, B.Y., Mian, A.S., Liu, W., Krishna, A.: Using kinect for face recognition under varying poses, expressions, illumination and disguise. In: Applications of Computer Vision (WACV), 2013 IEEE Workshop on. pp. 186–192. IEEE (2013)

18. Lu, J., Wang, G., Moulin, P.: Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning. In: International Conference on Computer Vision (ICCV), 2013 IEEE Conference on (2013)
19. Oja, E.: Subspace methods of pattern recognition, vol. 4. Research Studies Press England (1983)
20. Ortiz, E., Wright, A., Shah, M.: Face recognition in movie trailers via mean sequence sparse representation-based classification. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 3531–3538 (2013)
21. Ross, D.A., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. International Journal of Computer Vision 77(1-3), 125–141 (2008)
22. Shakhnarovich, G., Fisher, J.W., Darrell, T.: Face recognition from long-term observations. In: European Conference on Computer Vision (ECCV), pp. 851–865. Springer (2002)
23. Uzair, M., Mahmood, A., Mian, A., McDonald, C.: A compact discriminative representation for efficient image-set classification with application to biometric recognition. In: Biometrics (ICB), 2013 International Conference on. IEEE (2013)
24. Viola, P., Jones, M.J.: Robust real-time face detection. International journal of computer vision 57(2), 137–154 (2004)
25. Wang, R., Chen, X.: Manifold discriminant analysis. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 429–436. IEEE (2009)
26. Wang, R., Guo, H., Davis, L.S., Dai, Q.: Covariance discriminative learning: A natural and efficient approach to image set classification. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2496–2503. IEEE (2012)
27. Wang, R., Shan, S., Chen, X., Gao, W.: Manifold-manifold distance with application to face recognition based on image set. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. pp. 1–8. IEEE (2008)
28. Yamaguchi, O., Fukui, K., Maeda, K.i.: Face recognition using temporal image sequence. In: Automatic Face and Gesture Recognition (FG), 1998 IEEE International Conference on. pp. 318–323. IEEE (1998)
29. Yang, M., Zhu, P., Gool, L.V., Zhang, L.: Face recognition based on regularized nearest points between image sets pp. 1–7 (2013)
30. Zhu, P., Zhang, L., Zuo, W., Zhang, D.: From point to set: Extend the learning of distance metrics. In: International Conference on Computer Vision (ICCV), 2013 IEEE Conference on. IEEE (2013)