Towards Large-Scale Face Recognition Based on Videos

Meltem Yalcin, Hakan Cevikalp, Hasan Serhan Yavuz
Eskisehir Osmangazi University
Machine Learning and Computer Vision Lab, Eskisehir, Turkey
{yalcinmeltem26, hakan.cevikalp, hsyavuz}@gmail.com

Abstract

This paper introduces a new method to find the most important samples for classification in image sets to speed-up the classification phase and reduce the storage space for large-scale face recognition tasks that use image sets obtained from face videos. We approximate the image sets with the kernelized convex hulls and show that it is sufficient to use only the samples that participate to shape the image set boundaries in this setting. To find those important samples that form the image set boundaries in the feature space, we employed the kernelized Support Vector Data Description (SVDD) method which finds a compact hypersphere that fits the image set samples best. Then, we show that these kernelized hypersphere models can also be used to model image sets for classification purposes. Lastly, we introduce ESOGU-285 (ESkisehir OsmanGazi University) Face Videos database that includes 285 people since the most popular video datasets used for set based recognition methods include either a few amount of people or large amount of people with just a few (or single) video collections. The experimental results on small sized standard datasets and our new larger sized dataset show that the proposed method greatly improves the testing times of the classification system (we obtained speed-ups up to a factor of 10 in ESOGU Face Videos dataset) without a significant drop in accuracies.

1. Introduction

Face/object recognition based on image sets has been attracting more attention in recent years owing to the fact that collecting a set of images for recognizing people/objects becomes increasingly convenient and easy with the popularization of video cameras and cell phone cameras. In this setup, the user supplies a set of images of the same unknown individual rather than supplying a single query image. In general, the gallery also contains a set of images for each known individual, so the system must recover the individual whose gallery set is the best match for given query set. As a result, the image set recognition task naturally arises in a wide range of contexts including video-based recognition, surveillance, and personal albums. The query and gallery sets may contain large variations in pose, illumination, and scale. For example, even if the images were taken on the same occasion, they may come from different viewpoints or from face tracking in surveillance video over several minutes.

Recognition methods using image sets generally outperform the ones for single instance based recognition, both because they incorporate information about the variability of the individual's appearance and because they allow the decision process to be based on comparisons of the most similar pairs of query and gallery images - or on local models based on these. Moreover, in many applications, image sets are also the most natural form of the input to the system since obtaining image sets does not generally require cooperation from the individuals. Recognition based on image sets offers these great advantages, but at the same time it poses new challenges since the traditional classification methods such as Support Vector Machines (SVMs) or k-nearest neighbor (k-NN) cannot be used directly in this setup.

Existing classification methods using image sets differ in the ways in which they represent the image sets and compute the distances (or similarity) between them. Some methods [1, 15] used parametric probability distributions to model image sets, and Kullback-Leibler divergence is used to measure the similarity between these distributions. However, as noted in [18, 2], these methods are not very robust when the test sets have only weak statistical relationships to the training ones. Nonparametric methods [20, 6, 8, 2, 18] use different models to approximate image sets. Yamaguchi et al. [20] used linear subspaces to model image sets and they used canonical angles between subspaces to measure the similarity between them. Cevikalp and Triggs [2] used linear/kernelized affine/convex hull models to approximate image sets and geometric distances (distances of closest approach) between these models are used to measure the similarity. This method can be seen as enhance-

ment of nearest neighbor classification that attempts to reduce its sensitivity to random variations in sample placement by "filling in the gaps" around the examples. Although still based on the closest-point idea, classification method using affine/convex hull models replaces point-to-point or point-to-model comparisons with training-model to test-model ones. This methodology offers a number of attractive properties: the model for each individual can be fitted independently; computing distances between models is straightforward due to the convexity; resistance to outliers can be incorporated by using robust fitting to estimate convex models.

After introduction of affine/convex hull models, different variants of these models have been proposed [9,21]. For example, SANP (Sparse Approximated Nearest Points) [9] methodology extended the affine hull method by enforcing the sparsity of samples used for affine hull combination, and reported good accuracies. However, this method is very complex in the sense that it requires setting 3 design parameters beside the affine hull model parameters. It is also slow since one has to solve a complex optimization problem that includes minimization of L1 norm of some vectors, which makes it unsuitable for real-time applications as verified in our experiments. In a similar manner, [21] used regularized affine hull (RAH) models to represent image sets where L2-norms of affine hull combination coefficients are minimized during computing the smallest distances between affine hulls. Although this requires solving a much easier optimization problem compared to [9], it is still not suitable for real-time applications. More recently, new extensions of these methods used so-called collaborative representations for affine/convex hull models [19,22]. The basic difference is that they model all gallery sets as a single affine/convex hull and then query set is classified by using the reconstruction residuals computed from only individual gallery sets. However, as we show below, these methods are bound to fail for large-scale applications. Other methods using sparse models for image set based recognition can be found in [5,4,3]. Most of the mentioned methods above have kernelized versions that can be used to estimate nonlinear face models.

There are also many methods that seek to build nonlinear approximations of the manifold of face appearances, typically embedding local linearity within a globally nonlinear model. For instance, Fan and Yeung [6] use hierarchical clustering to discover local structures and approximate each local structure with a linear subspace. Wang et al. [18] follow a similar approach and they use nearest neighbor clustering to find the local structures forming the nonlinear manifold. Wang and Chen [17] extends MMD method as the manifold discriminant analysis (MDA) to improve the between-manifold distances. Cevikalp and Triggs [2] use spectral clustering to find the local structures and model the local structures with affine subspaces. Hadid and Pietikainen [8] apply k-means clustering to find local structures and model each local structure with the cluster center. All these methods were inspired by the nonlinear manifold modeling approach of Roweis and Saul [13], but they replace the locally affine models with different models as described above.

Our Contributions: We consider large-scale face recognition applications using image sets collected from videos in this paper. We first discuss main challenges that will be encountered for such applications. Then, we question suitability of the existing methods in the literature for large-scale applications and then propose an efficient method that will make large-scale image set based recognition feasible. To this end, we propose a method to find the most essential samples in image sets for classification to reduce the image set samples. SVDD method, which finds a compact kernelized hypersphere that best fits the image set samples, is used to determine the most essential samples. In addition, we show that the kernelized hypersphere models can be used for set based face recognition. It should be noted that most popular video datasets used for set based recognition methods are not large-scale and they include only few person classes. Therefore, to test the proposed method, we developed a new video dataset, called ESOGU Face Videos, that includes 2280 videos belonging to 285 individuals. The total number of frames is about 764 K. Although this dataset cannot be considered as large-scale data, it was still sufficient to show that the most recent face recognition methods using image sets have serious drawbacks related to computational complexity or representation of image sets.

2. Challenges for Large-Scale Face Recognition Based on Image Sets

One of the biggest challenge of the large-scale set based face recognition systems will be related to saving all data on a computer. Since even short videos may include hundreds of frames, the face detectors will return many face images for a single person. Thus, one needs to find a sophisticated technique to reduce the amount of original data without causing significant drop in recognition performance. Reducing techniques using random selections can significantly decrease the performance as reported in the results of experiments, that is carried out with different number of face images [4,3,9]. The second challenge will be to choose good models to represent image sets. The most popular video datasets such as Honda/UCSD [10] or YouTube Celebrities [11] used for set based recognition have videos of people with different poses including frontal, left or right profiles and poses between those, thus the face images in a set construct a nonlinear manifold which is locally linear. Although methods that approximate these nonlinear

image sets with a single linear/affine subspace or linear convex hulls produce very high accuracies on current data sets, the performances of these methods will drop as the number of people in the gallery set increases since these models will seriously overestimate the true extent of the classes and introduce large overlapping regions between image sets, cf., illustration in Fig. 1. Thus, only kernelized versions of these methods or methods that build nonlinear face manifolds using linear models will give satisfactory performance. The last challenge will be the real-time performance of the recognition system. An efficient system must return the individual (among the thousands of people) whose face image set is the best match for the given query set in a reasonable time.

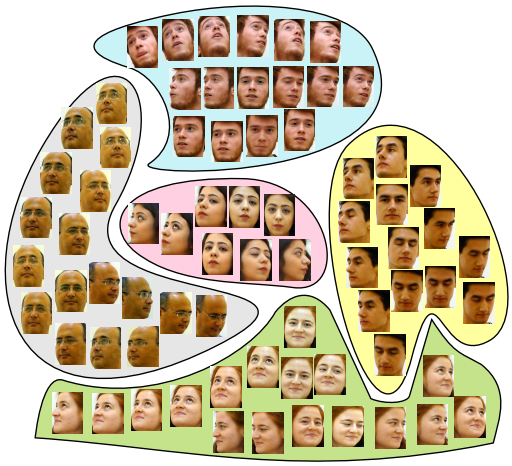

Figure 1. In large-scale applications, using either linear affine or convex hull models for representation of image sets causes large overlapping regions between these linear models. In this example, for affine hull model, all image sets span entire 2D plane thus it is impossible to separate all these sets; for convex hull model, most neighboring image sets have overlaps and it is only possible to separate furthest image sets.

In set based face recognition, the methods using the so-called joint or collaborative representations report very good accuracies on small sized datasets but they will likely to fail in large scale applications. In these methods, all gallery sets belonging to different individuals are approximated with a single combined affine/convex hull, and query set is classified by using the reconstruction residuals that only come from individual gallery sets. We adopt the illustration given in [22] to show how these methods get weak in large-scale applications. In case of a few image sets, one can model all image sets with a single convex hull and find the distances from the query sets to this convex hull as illustrated in Fig. 2 (a). But, when the number of person classes is large as in Fig. 2 (b), query sets will be typically inside the combined large convex hull built from all gallery sets. As a result, all distances from the convex hulls of query sets to this combined convex hull will become zero and the coefficients which will be used for computing residuals will be almost random. For affine hull models, the situation gets even worse since three independent face images (not sets) are enough to span all two-dimensional plane. Another problem would be related to the computational difficulty. Some collaborative representation based methods as in [19] require taking inverse of matrices with size $(n \times n)$, where n is the total number of images in the gallery. Large-scale applications result in very large matrices that would be very impractical to fit them into the memory, not to mention the difficulty of taking their inverses. Our experiments confirm this fact since we failed to implement some collaborative representation based methods because of memory issues on our moderate sized video dataset.

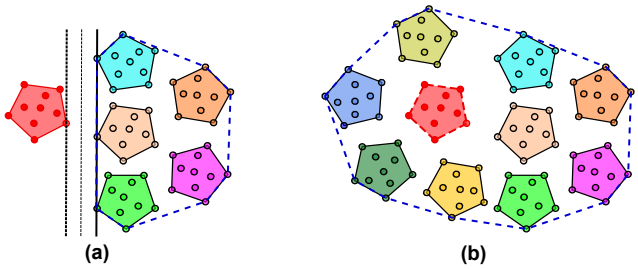

Figure 2. Comparisons of small and large-scale scenarios for set based recognition. In (a), the number of the image sets belonging to different people in gallery is small so one can model all gallery sets as a single convex hull and find the distance from the convex hull of the query set to this hull. But, when the number of people is increased, the query sets will be typically inside the convex hull formed by combining all image sets in the gallery as illustrated in (b). In such cases, the distances will become zero and the coefficients which will be used for computing residuals will be almost random. This will cause the collaborative model classifier to fail.

3. Proposed Method

In the proposed method, we approximate image sets with the kernelized convex hulls as in [2] since convex hulls are tighter models compared to affine hulls, and they provide better localization in large-scale applications. Let the face image samples be $\mathbf{x}_{ck} \in \mathbb{R}^d$, where $c = 1, \ldots, C$ indexes C image sets (individuals) and $k = 1, \ldots, n_c$ indexes n_c samples of image set c. Let $\phi(.)$ be the implicit feature space embedding and $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ be the corresponding kernel function, where $\langle . \rangle$ denotes the feature space inner product. A kernelized convex hull of samples $\mathbf{x}_{ck}$ is defined as

$$H_c^{convex} = \{\phi(\mathbf{x}) = \textstyle\sum_{k=1}^{n_c} \alpha_{ck} \phi(\mathbf{x}_{ck}) | \sum_{k=1}^{n_c} \alpha_{ck} = 1,\ 0 \leq \alpha_{ck} \leq 1\}. \quad (1)$$

If we set the upper bound on the convex combination coefficients to values, U, smaller than 1, several samples need to

be activated to ensure $\sum_{k=1}^{n_c} \alpha_{ck} = 1$, giving a more compact convex approximation that lies strictly inside the convex hull of the samples. This trick provides more robustness against to outliers during computation of the distances between convex hulls.

Given two compact kernelized convex hulls, the geometric distance between them can be found by solving the following constrained convex quadratic programming (QP) problem,

$$\begin{aligned} \underset{\boldsymbol{\alpha}_i, \boldsymbol{\alpha}_j}{\arg\min} \;\; & \left\| \Phi(\mathbf{X}_i)\boldsymbol{\alpha}_i - \Phi(\mathbf{X}_j)\boldsymbol{\alpha}_j \right\|^2 \\ \text{s.t.} \;\; & \sum_{k=1}^{n_i} \alpha_{ik} = \sum_{\tilde{k}=1}^{n_j} \alpha_{j\tilde{k}} = 1, \;\; 0 \leq \alpha_{ik}, \alpha_{j\tilde{k}} \leq U, \end{aligned} \tag{2}$$

where $\Phi(\mathbf{X}_i) = [\phi(\mathbf{x}_{i1}), \ldots, \phi(\mathbf{x}_{in_i})]$ represents the matrix whose columns are the mapped samples of set i, and $\boldsymbol{\alpha}_i$ is a vector containing the corresponding α_{ik} coefficients. It should be noted that the objective function of (2) can be written as $\boldsymbol{\alpha}^\top \mathbf{K} \boldsymbol{\alpha}$ by setting $\Phi(\mathbf{X}) = [\Phi(\mathbf{X}_i) \;\; -\Phi(\mathbf{X}_j)]$ and $\boldsymbol{\alpha} \equiv \binom{\boldsymbol{\alpha}_i}{\boldsymbol{\alpha}_j}$, where $\mathbf{K} = \Phi(\mathbf{X})^\top \Phi(\mathbf{X})$. This problem is closely related to the classical SVM classifier formulation, which finds a separating hyperplane between two convex hulls based on exactly the same pair of closest points. Thus, the same problem can also be solved by training an SVM that separates the query set from the given gallery one as explained in [2].

It is well-known that the solution returned by an SVM classifier is sparse and completely determined by the samples that are near the decision boundaries where the rival class samples approach to each other (these samples are called the support vectors), and all other samples far from these regions do not contribute to the solution. Therefore, the support vectors are the "essential" training points for classification and the goal of the SVM training is to discover them. If we generalize this rule for arbitrary convex sets, the geometric distances between them will always be determined by the samples in the vicinity of image sets' outer boundaries. Thus, if we can find the samples forming the image set boundaries in the feature space, we can ignore the remaining samples. This will greatly reduce the required disk storage space because only the relevant data needed for image set classification will be saved, and will significantly improve the testing speed since one has to solve smaller sized QP problems.

Kernelized one-class classifiers [16, 14] can be used to determine the samples that shape the image set boundaries. Both methods yield to the similar results for certain kernel function types such as the Gaussian kernels, but their solutions are different for the linear case if the data is not preprocessed to have unit norm. Therefore, we prefer to use the SVDD method of Tax and Duin [16] since the geometrical intuition behind the method is very similar to our goal. SVDD method aims to find a closed boundary around the data. To this end, it finds a compact bounding hypersphere where the most of data samples lie in that hypersphere.

The bounding hypersphere of a point set $\left\{ \mathbf{x}_k \in \mathbb{R}^d | k = 1, \ldots, n \right\}$ is characterized by its center $\mathbf{s}$ and radius r. These can be found by solving the quadratic programming problem

$$\begin{aligned} \underset{\mathbf{s}, r \geq 0, \boldsymbol{\xi} \geq 0}{\arg\min} \;\; & \left(r^2 + \gamma \sum_k \xi_k \right) \\ \text{s.t.} \;\; & \|\mathbf{x}_k - \mathbf{s}\|^2 \leq r^2 + \xi_k, \quad k = 1, \ldots, n, \end{aligned} \tag{3}$$

or its dual

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\arg\min} \;\; & \left(\sum_{k,l} \alpha_k \alpha_l \langle \mathbf{x}_k, \mathbf{x}_l \rangle - \sum_k \alpha_k \|\mathbf{x}_k\|^2 \right) \\ \text{s.t.} \;\; & \sum_k \alpha_k = 1, \;\; \forall k \;\; 0 \leq \alpha_k \leq \gamma. \end{aligned} \tag{4}$$

The α_k are Lagrange multipliers and $\gamma \in [1/n, 1]$ is the ceiling parameter that can be set to a value less than one to reduce the influence of outliers. The objective function is convex so a global minimum exists. In the kernelized case, we have to replace all inner products $\langle \mathbf{x}_k, \mathbf{x}_l \rangle$ with kernel evaluations $k(\mathbf{x}_k, \mathbf{x}_l)$. The dual formulation typically yields a sparse solution in terms of the support vectors (samples that correspond to the nonzero Lagrange multipliers), and they come from object class boundaries when nonlinear kernel functions are used as illustrated in Fig. 3. In our setup, we solve the QP problem (4) for each image set in the gallery offline and keep only these support vectors for each set. During testing, we run the same algorithm for the query set and compute the kernelized convex hull distances by using these reduced image sets that contain only the essential support vectors. In this way, the amount of data is greatly reduced without any significant decrease in the accuracy, and the testing time is greatly shortened since smaller sized QP problems are solved. We call the proposed method as Reduced Convex Hull based Image Set Distance (RCHISD) method.

It should be noted that we can find the center of the kernelized hypersphere model of the c-th class by using the nonzero $\boldsymbol{\alpha}^*$ coefficients returned by the QP solver as follows

$$\mathbf{s}_c = \sum_k \alpha_k^* \mathbf{x}_{ck}. \tag{5}$$

The corresponding radius is $r_c = ||\mathbf{x}_{ck} - \mathbf{s}_c||$ for any $\mathbf{x}_{ck}$ for which $0 < \alpha_k^* < \gamma$. Therefore, we can also find the most similar gallery set to the given query by using the distances between the kernelized hypersphere models. The geometric distance between two kernelized hyperpsheres, hs_c and hs_q

(characterized by their center and radius), is given as

$$d(hs_c, hs_q) \;=\; ||\mathbf{s}_c - \mathbf{s}_q|| - (r_c + r_q), \qquad (6)$$

where

$$||\mathbf{s}_c - \mathbf{s}_q|| = \sqrt{\sum_{i,j} \alpha_{ci}\alpha_{cj} \langle \mathbf{x}_{ci}, \mathbf{x}_{cj} \rangle - 2\sum_{i,k} \alpha_{ci}\alpha_{qk} \langle \mathbf{x}_{ci}, \mathbf{x}_{qk} \rangle + \sum_{k,l} \alpha_{qk}\alpha_{ql} \langle \mathbf{x}_{qk}, \mathbf{x}_{ql} \rangle}.$$

One needs to use a few support vectors that correspond to the nonzero Lagrange multipliers to compute the above distance, thus this computation is too fast. In our experiments, we also used hypersphere models for image set classification and compared these results to the ones obtained by the kernelized convex hulls. To the best our knowledge, this is the first time of the use of the hypersphere models for image set classification.

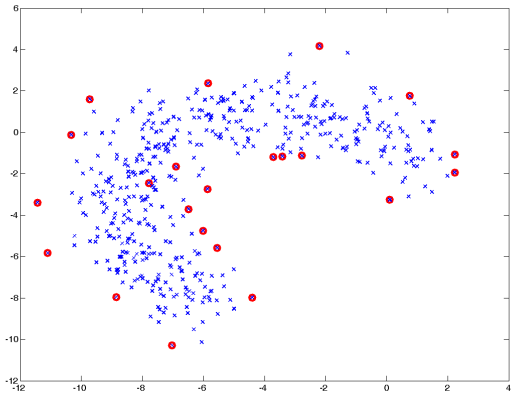

Figure 3. Nonlinearly distributed data and the support vectors (shown with red circles around the data samples) returned by SVDD using a Gaussian kernel. Support vectors come from the object boundaries when the Gaussian kernel width is set properly.

In the kernelized case of the proposed method, there are two parameters: the ceiling parameter (γ) and the Gaussian kernel width which define the reduced sets. As we mention in the experimental study section, the results are not very sensitive to γ, but the Gaussian kernel width highly affects the number of reduced elements in the set. So, the Gaussian kernel width parameter is important for determining the size of the reduced image sets. The smaller values of this parameter return more support vectors whereas larger values return less support vectors as demonstrated in Fig. 4. In this example, we gradually increase the value of the Gaussian kernel width from 1.5 to 5.0 and plot the returned support vectors with the red circles around the data samples. The more data yields better accuracies but the less data is faster during testing. When the parameter is adjusted properly, lower amount of data are used to represent the image sets collected from videos, which results in less amount of storage and better classification speeds.

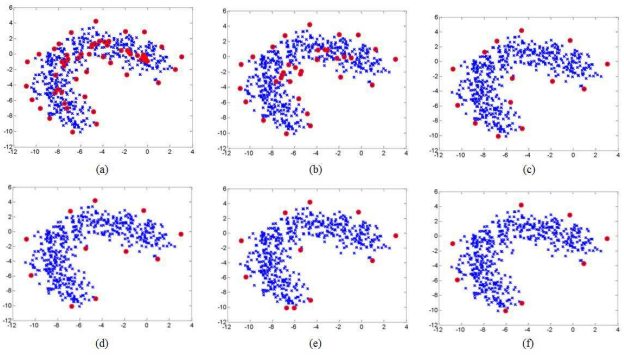

Figure 4. Support vectors (red circles) returned by SVDD algorithm for different parameters of Gaussian kernel width: (a) kernel width is set to 1.5, (b) kernel width is set to 2.0, (c) kernel width is set to 2.5, (d) kernel width is set to 3.5, (e) kernel width is set to 4.5, (f) kernel width is set to 5.0. Note that less support vectors are returned as the kernel width is increased.

4. Experiments

In this study, we tested recognition accuracies and testing speeds of the proposed methods on two popular small sized databases CMU MoBo [7] and Honda/UCSD [10] and our new larger sized ESOGU Face Videos data set. To allow comparison with the literature on various datasets, we followed the simple protocol of [2, 21, 18, 19]: the face images detected from video frames were histogram equalized but no further pre-processing such as alignment or background removal was performed on them. In our database, we tested both gray-level values and local binary pattern (LBP) features for classification. For affine hull methods, subspace dimensions are set by retaining enough leading eigenvectors to account for 98% of the overall energy in the eigen-decomposition. For all kernelized methods we used the Gaussian kernels and the Gaussian kernel width is determined based on experiments using randomly selected subsets of image sets. We compared the proposed method RCHISD (Reduced Convex Hull based Image Set Distance) to the linear/kernelized affine hull method (AHISD) [2], linear/kernelized convex hull method (CHISD) [2], Mutual Subspace Method (MSM) [20], SANP [9], Regularized Nearest Points (RNP) [21], Collaboratively Regularized Nearest Points(CRNP) [19], Manifold-Manifold Distance (MMD) [18], and Self-Regularized Nonnegative Adaptive Distance Metric Learning (SRN-ADML) [12]. In addition to these methods, we also tested linear/kernelized bounding hypersphere (HS) models for image set classification.

4.1. Experiments on MoBo Data Set

The MoBo (Motion of Body) data set contains 96 image sequences of 24 individuals walking on a treadmill. The images were collected from multiple cameras under four different walking situations: slow walking, fast walking, in-

cline walking, and carrying a ball. Each image set includes both frontal and profile views of the subjects faces. We used LBP feature sets from [2].

As in [2], we randomly select one image set from each class for the gallery and used the remaining 3 for testing. This was repeated 10 times and we report averages of the classification rates over the 10 runs. Table 1 shows the accuracies and testing times for all tested methods. Testing time shows the amount of time spent to classify a single test set on the average. We tested the kernelized reduced convex hull classifiers for different ceiling parameter γ values changing between 0.10 and 1, and all of them returned same accuracies. So, we conclude that the results are not very sensitive to this parameter and we fix it to $\gamma = 0.20$ for all of the experiments. The linear/kernelized convex hull models and SANP achieve the best results and reducing the image sets samples by using SVDD does not impact the accuracy significantly, but improves the testing time. When reduced image sets are tested by computing pair-wise distances independently, testing time of a query is approximately 4 times faster compared to using full image sets. The linear hypersphere method is the worst performing method, but it is one of the fastest methods. Similarly, the accuracy of the kernelized hypersphere method is also low compared to other kernelized methods, but it is the fastest one among all tested kernelized methods. Regarding the reduced set size, the total number of face images in all sets is 48789 and it is reduced to 7098 by using SVDD method without a significant drop in accuracy. Kernel AHISD and SANP methods are the worst methods in terms of testing time. It should be noted there is not a significant accuracy difference between the linear and kernelized versions of all methods except for HS classifier since the number of the people in the dataset is small.

Table 1. Classification Rates (%) and Testing Times on the MoBo Dataset.

Method	Accuracy	Testing Time (sec)
Linear AHISD	95.3 ± 2.6	32.0 sec
Linear CHISD	$\mathbf{98.1} \pm 0.9$	25.6 sec
Linear HS	71.9 ± 4.7	0.6 sec
MSM	92.4 ± 1.9	9.2 sec
SANP	$\mathbf{98.1} \pm 0.9$	40.2 sec
RNP	93.8 ± 2.7	11.3 sec
CRNP	97.4 ± 0.8	15.8 sec
SRN-ADML	95.3 ± 1.6	30.0 sec
MMD	94.7 ± 2.3	10.6 sec
Kernel AHISD	96.4 ± 2.5	87.3 sec
Kernel CHISD	$\mathbf{98.1} \pm 0.9$	32.8 sec
Kernel HS	87.8 ± 2.8	5.8 sec
Kernel RCHISD	97.3 ± 1.3	8.3 sec

4.2. Experiments on Honda/UCSD Data Set

The Honda/UCSD data set [10] was collected for video-based face recognition. It consists of 59 video sequences involving 20 individuals. Each sequence contains approximately 300-500 frames. Twenty sequences were set aside for training, leaving the remaining 39 for testing. The detected faces were resized to gray-scale images and histogram equalized, and the resulting pixel values were used as features. Table 2 shows the accuracies and testing times for all tested methods. We tested the kernelized convex hull classifiers for different ceiling parameter values changing between 0.10 and 1, and all of them returned the same accuracy of 100%. The kernelized convex hull models achieve the best results and reducing the image sets samples by using SVDD does not impact the accuracy but improves the testing time. When reduced image sets are tested by computing pair-wise distances independently, testing time of a query is 2 times faster compared to using full image sets. The linear hypersphere method is the worst performing method, but it is one of the fastest methods. Similarly, the accuracy of the kernelized hypersphere method is also low compared to other kernelized methods, but it is the fastest one among all tested kernelized methods.

Table 2. Classification Rates (%) and Testing Times on the Honda/UCSD Dataset.

Method	Accuracy	Testing Time (sec)
Linear AHISD	97.4	1.6 sec
Linear CHISD	97.4	5.1 sec
Linear HS	59.0	0.6 sec
MSM	97.4	2.14 sec
SANP	97.4	16.7 sec
RNP	**100**	5.4 sec
CRNP	**100**	2.6 sec
SRN-ADML	97.4	6.18 sec
MMD	**100**	7.11 sec
Kernel AHISD	97.4	14.2 sec
Kernel CHISD	**100**	7.6 sec
Kernel HS	94.9	2.8 sec
Kernel RCHISD	**100**	3.7 sec

4.3. Experiments on ESOGU-285 Face Videos Dataset

ESOGU Face Videos dataset includes videos of 285 people captured in two sessions separated by at least three weeks. In each session, we captured four short videos with four different scenarios for each person. In the first scenario, the subjects are asked to make free head movements under normal illumination conditions similar to video recordings in Honda/UCSD. In the second one, the subjects pretend to talk on a cell phone with free head movements.

The third and the last videos include recordings of free head movements when the subjects are illuminated from the right and left, respectively. Some frames from videos are shown in Fig. 5. We manually cropped the faces using a semi-automatic annotation tool. We used both (40×30) gray-level and LBP values as visual features.

We used the image sets captured in the first session as gallery set and the sets captured in the second session for testing. The recognition accuracies and testing times are given in Table 3. We could not implement CRNP because of memory issues since it requires to operate on matrices with size $n \times n$, and $n = 410251$ is the number of frames in the gallery (In table 2, OOM indicates the "out of memory" problem). We would like to point out that SANP method is very slow for gray-level values. However, the best accuracy for LBP values is obtained by SANP followed by Kernel CHISD and MMD whereas the Kernel CHISD is the best performing method alone for gray-level features. The total number of frames in both gallery and query sets is 764006 and it is reduced to 104716 for LBP and 149520 for gray level features by using kernelized SVDD. Our proposed Kernel RCHISD method, which uses the reduced sets, achieves similar results to Kernel CHISD, but it is approximately 10 times faster for LBP and 6 times faster for gray-level features. The linear hypersphere is again one of the fastest methods but it has the worst recognition accuracy.

As opposed to the results on small sized datasets, there is a big difference between classification accuracies of linear methods and their kernelized counterparts especially for gray-level values. More precisely, both kernelized convex hull and affine hull models achieve much higher accuracies than linear methods for gray-level values as expected. In a similar manner, the kernelized affine hull model significantly outperforms the linear affine hull model for LBP features, but there is not a significant performance difference between the linear and kernelized convex hull models since convex hull is a much tighter model compared to affine hulls (however we should expect a difference if we further increase the number of people). These results also indicate that LBP features are more discriminative features compared to gray-level values and they yield to more compact face manifolds.

5. Conclusion

In this work, we developed image set based classification methods which use the reduced image samples in each set to lessen the required storage space and speed-up the testing process for large-scale face recognition applications. To this end, we showed that we need to keep only the image sets samples that form the image set boundaries when kernelized convex hulls were used to approximate image sets. SVDD method, which finds a compact hypersphere that fits the image set samples best, is used to determine the samples forming image set boundaries. Experimental results verify that reducing image set samples via SVDD greatly improves the testing time without a significant drop in accuracy.

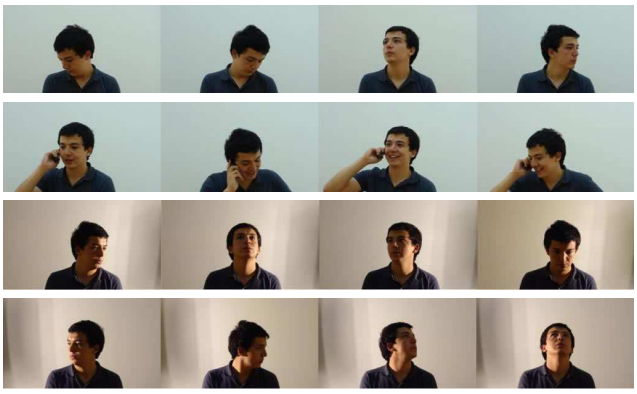

Figure 5. Some frames selected from videos captured in each session. The first row shows the recording of free head movements without illumination, the second row shows the recording of phone call, and the third and the last rows show the recordings of free head movements when the subjects are illuminated from the right and left directions.

Another contribution of the study is the investigation of the suitability of the hypersphere models for approximating image sets. Experiments show that hypersphere models yield to lower accuracies compared to affine or convex hull models, but they are extremely faster. Therefore, these models can be used to return the nearest approximate candidates of the gallery sets to the given query set quickly, and then a more accurate similarity search can be done among the returned candidate sets by using affine/convex hull approximations.

Lastly, it was shown that accuracies of methods using linear models to approximate image sets drop as the number of classes is increased. Especially it has been observed a significant accuracy drop when looser linear models such as affine hulls and linear subspaces are used to approximate image sets. We also verified that some recently proposed collaborative methods and methods using sparse models cannot be applied to large-scale data (even to moderate size data) due to the memory or speed problems. It should also be noted that the proposed methods are not limited with face recognition. They can be used in other visual recognition problems where each example is represented by a set of images.

Acknowledgments: This work was funded by the Scientific and Technological Research Council of Turkey (TUBİTAK) under Grant numbers EEEAG-114E014 and EEEAG-113E118.

Table 3. Classification Rates (%) and Testing Times on the ESOGU-285 Face Videos Dataset.

Methods	Grayscale Values		LBP Features	
	Accuracy	**Testing Time**	**Accuracy**	**Testing Time**
Linear AHISD	44.30	22.0 sec	66.75	180.0 sec
Linear CHISD	**55.09**	179.6 sec	76.58	390.1 sec
Linear HS	29.04	3.9 sec	39.47	0.8 sec
MSM	50.08	2.3 sec	69.56	5.1 sec
SANP	51.92	29771 sec	**79.12**	564.6 sec
RNP	46.66	1731.7 sec	51.92	2205.3 sec
CRNP	OOM	-	OOM	-
SRN-ADML	45.35	364.6 sec	68.42	380.2 sec
MMD	52.02	7.2 sec	**77.63**	30.4 sec
Kernel AHISD	62.11	2015.0 sec	76.05	4369.0 sec
Kernel CHISD	**62.19**	233.3 sec	**77.63**	480.4 sec
Kernel HS	43.68	61.9 sec	49.39	12.9 sec
Kernel RCHISD	61.23	39.7 sec	75.36	46.1 sec

References

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

[2] H. Cevikalp and B. Triggs. Face recognition based on image sets. In *CVPR*, 2010. 1, 2, 3, 4, 5, 6

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

[4] Y.-C. Chen, V. M. Patel, S. Shekhar, R. Chellappa, and P. J. Phillips. Video-based face recognition via joint sparse representation. In *Automatic Face and Gesture Recognition Conference*, 2013. 2

[5] Z. Cui, H. Chang, S. Shan, B. Ma, and X. Chen. Joint sparse representation for video-based face recognition. *Neurocomputing*, 135:306–312, 2014. 2

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

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

[8] A. Hadid and M. Pietikainen. From still image to video-based face recognition: an experimental analysis. In *International Conference on Automatic Face and Gesture Recognition*, 2004. 1, 2

[9] Y. Hu, A. S. Mian, , and R. Owens. Face recognition using sparse approximated nearest points between image sets. *IEEE Transactions on PAMI*, 34(3):1992–2004, 2012. 2, 5

[10] K. C. Lee, J. Mo, M. H. Yang, and D. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *CVPR*, 2003. 2, 5, 6

[11] V. P. M. Kim, S. Kumar and H. Rowley. Face tracking and recognition with visual constraints in real-world videos. In *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008. 2

[12] A. Mian, Y. Hu, R. Hartley, and R. Owens. Image set based face recognition using self-regularized non-negative coding and adaptive distance metric learning. *IEEE Transactions on Image Processing*, 22:5252–5262, 2013. 5

[13] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2319–2323, 2000. 2

[14] B. Schölkopf, J. Platt, and R. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471, 2001. 4

[15] G. Shakhnarovich, J. W. Fisher, and T. Darrell. Face recognition from long-term observations. In *ECCV*, pages 851–868, 2002. 1

[16] D. M. J. Tax and R. P. W. Duin. Support vector data description. *Machine Learning*, 54:45–66, 2004. 4

[17] R. Wang and X. Chen. Manifold discriminant analysis. In *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009. 2

[18] R. Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image sets. In *CVPR*, 2008. 1, 2, 5

[19] Y. Wu, M. Minoh, and M. Mukunoki. Collaboratively regularized nearest points for set based recognition. In *BMVC*, 2013. 2, 3, 5

[20] O. Yamaguchi, K. Fukui, and K.-I. Maeda. Face recognition using temporal image sequence. In *International Symposium of Robotics Research*, pages 318–323, 1998. 1, 5

[21] M. Yang, P. Zhu, L. V. Gool, and L. Zhang. Face recognition based on regularized nearest points between image sets. In *Automatic Face and Gesture Recognition Conference*, 2013. 2, 5

[22] P. Zhu, W. Zuo, L. Zhang, S. C.-K. Shiu, and D. Zhang. Image set-based collaborative representation for face recognition. *IEEE Transactions on Information Forensics and Security*, 9:1120–1132, 2014. 2, 3