ABSTRACT

Title of Document:	A WATCH-LIST BASED CLASSIFICATION SYSTEM.
	Ankit Jain, Master of Science, 2013
Directed By:	Professor Rama Chellappa, Department of Electrical and Computer Engineering

Watch-list-based classification and verification is advantageous in a variety of surveillance applications. In this thesis, we present an approach for verifying if a query image lies in a predefined set of target samples (the watch-list) or not. This approach is particularly useful at identifying a small set of target subjects and therefore can render high levels of accuracy. Further, this approach can also be extended to identify the query image exactly out of the target samples. The three-stages approach proposed here consists of using a combination of color and texture features to represent the image and further using, Kernel Partial Least Squares for dimensionality reduction followed by a classifier. This approach provides improved accuracy as shown by experiments on two datasets.

A WATCH-LIST BASED CLASSIFICATION SYSTEM

By

Ankit Jain

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Masters of Science
2013

Advisory Committee:
Professor Rama Chellappa, Chair
Professor K.J. Ray Liu
Professor Min Wu

Acknowledgements

First and Foremost, I would like to thank my advisor Professor Rama Chellappa for giving me this opportunity to work under his guidance. I am very grateful to him for his help and support which made my stay here at University of Maryland exciting and challenging. I would like to thank Professor K.J Ray Liu and Professor Min Wu for being on my defense committee and sparing their invaluable time reviewing my thesis. I would also like to thank Jaishankar Pillai for assisting me in all the scientific and technical issues that arose during this work and Raviteja Vemulapalli for all the endless discussions that helped me sail through some tough times.

I would like to express my gratitude towards my parents and my brother for their love and support. Lastly I would like to thank all for making my stay here memorable and above all I would like to thank GOD.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

A watch-list consists of a group of target samples that have some common attributes, which distinguishes them from the rest of the population, the non-target samples. For any watch-list based surveillance task, the recognition scheme first should find out if the query image is present in the watch-list or not, and if found the scheme should identify that individual. Here, I first present a watch-list based system from a person re-identification point of view and then formulate it as a set-based recognition system.

1.1 Person Re-identification

In a large public space covered by multiple non overlapping cameras, the person re-identification system tries to find the same person in some another view at some other arbitrary location. This problem is particularly difficult to solve due to large variations in visual appearance of the person arising out of changes in pose, lightning conditions and occlusion of the person. This requires a careful choice of features which can aid in addressing the large intra and inter class variations. Many approaches to solve this problem employ color and rectangular region histograms and other multiple feature-based representations. But these approaches are plagued with some problems. These methods assume that the gallery set and the query set contain the same images which might not be the case in a very large public setting in which the gallery set might include images whose labels are not known and which is much larger than the query set. This gives rise to set-based classification [1], in which the query image has to be checked only against the target set.

Figure 1.1 Highliting appearance changes of the same person

Figure 1.1 shows the large intra and inter class variations present in outdoors. Each column corresponds to the image of the same person taken from a different view. Changes in the appearance are due to pose, lighting and occlusion.

1.2 Set based Recognition

Set-based Recognition reduces the problem of verifying a small set of target subjects for the given query image rather than matching the query image for each individual subject. Considering the actual surveillance problem, it is very difficult for a fully automated system to identify with a good level of accuracy the subjects present on a watch-list without the help of human operators. Therefore, it makes more sense to consider the problem of matching a small target set where the true target samples are mostly included in the watch-list in addition to human intervention to accurately identify the true match of the query image. Thus, as in [1] we present a multi-shot verification approach to identify the query image as being present in the set followed by a one shot verification to identify the query image exactly. But in contrast to [1],

our approach separates the target and non-target samples in a reduced dimension space.

Figure 1.2 Target (watch-list) vs. Non-Target samples

As shown in Figure 1.1, the watchlist consists of target samples to which the query image is matched and the rest of the population forms the non-target samples. The images in the non-target set are unlabeled images of subjects which help in extracting discriminant information between target and non-target sets.

1.3 Feature Extraction

We adopt a combination of color and texture features as in [2]. RGB, YCbCr and HSV color-space give rise to three different color channels. Various combinations of Schmid and Gabor feature channels give rise to nineteen texture feature channels. The details of the features can be found in Chapter 3.

1.4 Partial Least Squares (PLS) vs. Rank SVM

The Rank SVM approach works by dividing the images into relevant and irrelevant image pairs coming from the target and the non- target set [1]. One of the drawbacks of this approach is that it employs the classifier on all pair of images from the non-target set, and the target set which can be really huge if the data comes from a public space dataset, where the non-target population is considerably large thereby giving way to a large number of constraints in the SVM formulation. Another drawback is that if a new person has to be added to the target-set, the formulation has to be computed again which can be very expensive. On the other hand the Partial least Squares (PLS) approach used in this thesis, works by first reducing the dimensions of the images by bringing together the target and the non- target set. This helps in two ways. First, the working dimensionality of the images to be used for the classifier is reduced considerably and second if we get a new target image to be included in the watch-list, this image can be directly subjected to the low-dimensional space without the requirement of computing the space again. Extending the concept of PLS to its kernel variant gives accurate and better results than RankSVM since it employs non-linear dimensionality reduction to separate the two sets in the low dimensional-subspace. The details of PLS and Kernel PLS are covered in Chapter 4.

1.5 Outline of the Thesis

The thesis is organized as follows. Previous works on person re-identification and set-based verification are discussed in Chapter 2. Chapter 3 presents the details of the feature extraction step on the images. Chapter 4 presents a detailed description of PLS and its kernel variant extensions. Chapter 5 formulates the learning problem using Kernel PLS. Chapter 6 gives a description of the datasets used. Chapter 7 explains in detail the experiments and results and finally Chapter 8 concludes the thesis with some suggestions for future work.

Chapter 2: Previous Work

The problem we address in this thesis is similar to the person re-identification problem which aims at matching people across disjoint camera views in a multi-camera system [3,2,4,5,6,7]. This type of recognition problem is faced with some bottlenecks. In a busy uncontrolled environment monitored by cameras from a distance, person verification relying upon biometrics such as face or gait is not robust. Secondly, as the transition time between different cameras varies from individual to individual, we cannot impose accurate temporal or spatial constraints. Also, the visual appearance features extracted from clothing and shape of people are not sufficient for recognizing people. This is because in winters people wear warm clothes such as winter jackets and everyone appears the same. Finally, a person's appearance undergoes large variations across non-overlapping camera views which lead to changes in view angle, lighting, background clutter and occlusion and hence different people across various views appear similar.

Many of the current person re-identification techniques try to seek a more distinctive and reliable feature for representing the person's appearance. These features range from color histograms [2,3], graph models [8], the spatial co-occurrence representation model [4], principal axis [5], rectangle region histograms [9], part-based models [10,11] to a combination of multiple features [2,6]. After feature extraction, many of these methods use l-1 norm [4], l-2 norm [5] or the Bhattacharya Distance [2] for classification. This distance-based model treats all features equally without discarding bad features selectively in each individual. Many

of the existing works focus on the problem of feature extraction and representation by a bag of words consisting of color and textual features. In addition to this matching, many existing works also exploit contextual cues. In [12, 13, 14], a brightness transfer function is introduced to explicitly compensate for lighting changes between cameras which increase the cost since segmenting each person from the image is costly. By modeling the transition time between two camera views, one can reduce the number of potential matches while using the probability distribution of transition time as a feature vector as in [15,16,17,18]. But as pointed out in [19], the transition time can be unreliable when a large number of moving objects is involved. In [2], the authors propose a boosting approach based on Adaboost to select a subset of optimal features for matching people. But as pointed out in [19], such a selection of features may not be globally optimal.

As in [1], many methods for re-identification treat this problem as a traditional image retrieval or recognition problem. It is assumed that both gallery sets and probe sets contain the same objects. But in a more realistic public environment, the gallery set is much bigger than the probe set and it becomes rather difficult to match against everybody in the gallery set exhaustively. In unconstrained public spaces as the number of people increase, their re-identification accuracy decreases significantly [7,20]. In such a setting, a person's appearance is only relatively stable for a short period of time. This restriction makes the problem even more challenging from other classification and verification tasks such as face recognition as the number of samples available for recognition is very limited. Therefore a model learned for matching two images of the same person can be easily over fitted.

Drawing parallels with the pedestrian tracking approach, we know that when the camera is fixed and the number of target persons to be recognized is small, the pedestrians can be easily tracked. As the size of the tracking system grows, it presents a more challenging problem because of the lack of temporal constraints when matching across non overlapping fields of view in a multi-camera network. In such a case if the viewpoint angle is known, we can perform classification [21]. For a problem which focuses only on the frontal viewpoint we could fit a triangular graph model [8] or a part-based model [22]. If multiple overlapping cameras are available, a panoramic appearance map [23] is preferred. Histograms are also useful for tracking [24] and pedestrian detection [25]. Correlations [12], spatial position in spatialograms [13], vertical position in principal axis [14] or scale in multi-resolution histograms [15] have also been used. A hybrid model constructed from the histogram and template appearance has also been discussed in [2]. We have also used a similar appearance-based model which is an ensemble of localized features consisting of a feature channel, location and binning information.

Once a discriminative appearance based model has been built, other learning methods such as Support Vector Machines (SVM) [30], k-nearest neighbors combined with SVM [31], decision trees [32], learning discriminative distance metrics [33] have also been exploited. Since feature augmentation results in a high-dimensional feature space, these methods cannot be directly used since they will have high computational requirements. Hence in such a case to reduce the dimensionality of the data, we use the Kernel Partial Least Squares [KPLS] in this work. The PLS technique has also been used in a slightly different setting in [34].

Machine learned ranking techniques such as RankSVM [35] and RankBoost [36] have been successfully used in text document analysis and information retrieval. In [1], the authors use primal RankSVM to solve the problem of person re-identification. Set-based verification based on RankSVM and PRDC was presented in [1]. As indicated in [1], the appearance of different subjects can be rather similar with large intra-class variations due to large changes in camera view and lighting conditions. Hence, it is difficult for a fully automated system to identify the true targets accurately and reliably without the help of a human operator. Thus, it is more relevant to consider the problem of re-identification as matching a small candidate set such that the true target candidates are mostly included therein. The set-based transfer learning framework for verification of each target person in [1] is based on the bipartite ranking model [19, 20]. The method in [1] explores the useful relative comparison between target and non-target data and makes use of this information to enhance the bipartite ranking analysis between target data. This concept of transfer learning aims to construct more robust statistical learning models that can benefit from the shared knowledge between related domains when training data is sparse and imbalanced. This work is one of the earliest in set-based verification using transfer learning approach. Next we explain the RankSVM approach used in [1] in detail.

In [1], the authors focus on three types of similarity and dissimilarity pairs which contribute to the bipartite ranking-based verification approach. The authors learn a score function f(x) which is a function of the difference vector x, computed as an absolute difference vector i.e. $f(x) = w^T x$ as a distance model. It says that the score function for a difference vector x^p computed from a relevant pair of images

should be greater than the difference vector x^n computed from a relevant pair of images where only one sample for computing x^n is one of the two relevant samples for computing x^p. Hence, the function ranking is expressed as $f(x^p) \geq \rho + f(x^n)$ where ρ is a non-negative margin variable. The authors propose three types of bipartite ranking model briefly summarized in the next paragraph.

The score comparison between a pair of relevant target person images and a related pair of irrelevant target pair images is denoted by red and green lines in Figure 2.1. Denote $O_i^t = (x_i^{t,p}, x_i^{t,n})$ where $x_i^{t,p}$ is the difference vector computed between a pair of relevant samples of the same target person and $x_i^{t,n}$ is the difference vector from a related pair of irrelevant samples. All such examples are denoted by the set $\{O_i^t\}$, so for the samples $\{O_i^t\}$, the authors take into consideration, the comparison $f(x^{t,p}) \geq \rho + f(x^{t,n})$. Further, they take into account the inter-class variations between any pair of target and non-target image. A score comparison between a pair of relevant target person images and a related pair of the irrelevant person images between the target person image and any non-target person image $O_i^{ts} = (x_i^{t,p}, x_i^{ts,n})$ where $x_i^{t,p}$ is defined as before and $x_i^{ts,n}$ is the difference vector between any sample for computing $x_i^{t,p}$ and any non-target person image sample. This set is further denoted by $\{O_i^{ts}\}$ and for these samples the following constraints need to be satisfied $f(x^{t,p}) \geq \rho + f(x^{ts,n})$. Another score comparison that the authors propose is between a pair of different target person images and a related pair of irrelevant person images between the target and any non-target $O_i^{tsn} = (x_i^{t,n}, x_i^{ts,n})$ where $x_i^{ts,n}$ is

the difference between one of the target images and between one of the target images for computing $x_i^{t,n}$ and any non-target person image. The set of all such images is denoted by O_i^{tsn} and the constraint to be satisfied for all such examples is $f(x^{t,n}) \geq \rho + f(x^{ts,n})$. Using the above three sets of images the authors develop two transfer bipartite ranking models namely RankSVM and Probabilistic Relative Distance Comparison (PRDC). The RankSVM optimization problem is presented as a score function $f(x) = w^T x$ as:-

$$\begin{aligned} &\min_w \frac{1}{2} \| w \|^2 \\ &w^T x_i^{t,p} \geq 1 + w^T x_i^{t,n}, \forall O_i^t \\ &w^T x_i^{t,p} \geq 1 + w^T x_i^{ts,n}, \forall O_i^{ts} \\ &w^T x_i^{t,n} \geq 1 + w^T x_i^{ts,n}, \forall O_i^{tsn} \end{aligned} \tag{2.1}$$

The solution for the above optimization is developed in [1] and Transfer PRDC is also explained in [1]. Figure 2.1 depicts the three cases as described above with the intra and inter class variations shown against the target and the non-target samples.

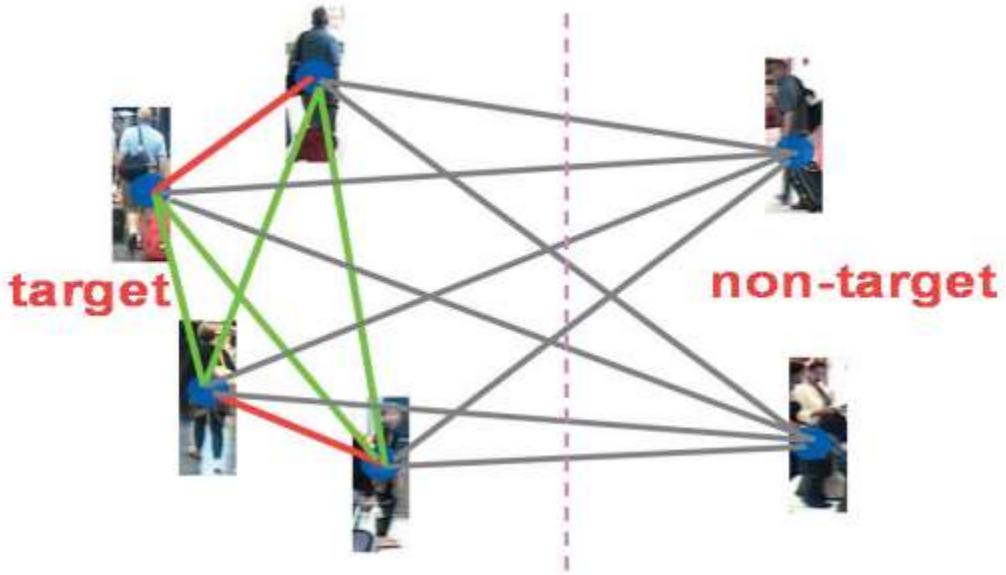

Figure 2.1 Variations among target and non-target data

We use Kernel PLS to tackle the above issue. The above optimization problem is computationally extensive since the number of constraints i.e. the number of relevant and irrelevant samples is large. We use Kernel PLS which takes into consideration the challenges presented in the above formulation and also gives better accuracy as discussed in Chapter 7. The use of Kernel PLS is motivated by the presence of high dimensionality, the small number of samples available to learn appearances and the high intra and inter class variation between the samples of the target and the non-target class.

Chapter 3: Feature Extraction

Although the proposed method can be applied to any set of features, in this thesis we use a histogram based-feature representation for each image. The feature vector is made of a mixture of color features (RGB, YCbCr and HSV color) and texture feature pattern (a combination of Schmid and Gabor features) as proposed in [1]. Each image is represented by a feature vector in a 2784 dimensional feature space. Next, we give a detailed description of the color and the texture features. First, the image of each person is divided into 6 horizontal stripes. These stripes capture the head, upper and lower torso and upper and lower legs. Next, for each stripe we obtain the RGB, YCbCr and HSV color features, Schmid and Gabor texture features and represent them as histograms. Each feature channel is represented by a 16-dimensional histogram vector (bin size) and in total there are 29 feature channels constructed for each stripe resulting in a 2784 dimensional vector for each image.

The 29 feature channels consist of 8 color channels (RGB, HS and YCbCr) and 21 texture features (Schmid [37] and Gabor [38]) applied to the luminance channel [2]. The parameters for Gabor filters for $\gamma, \lambda, \theta, \sigma^2$ were set to (0.3,0,4,2), (0.3,0,8,2), (0.4,0,4,1), (0.3,$\frac{\Pi}{2}$,8,2), (0.4, $\frac{\Pi}{2}$,4,1), (0.4, $\frac{\Pi}{2}$, 8 ,2). Further, the parameters τ, σ for the Schmid filter were set as (2,1), (4,1), (4,2), (6,1), (6,2), (6,3), (8,1), (8,2), (8,3), (10,1), (10,2),(10,3) and (10,4) respectively similar to [2].

3.1 Feature Channels

A feature channel for any image can be seen as a single channel transformation of a single image. This includes the eight-color channels from the color space and nineteen-texture channels. Each texture channel is a convolution with a filter and the luminance channel.

The Schmid filter helps in modeling invariance to viewpoint and pose (i.e. they are rotation invariant). They are defined as [2]:-

$$F(r,\sigma,\tau) = \frac{1}{Z}\cos\left(\frac{2\pi\tau r}{\sigma}\right)e^{-\frac{r^2}{2\sigma^2}} \tag{3.1}$$

Here r denotes the radius, Z is a normalizing constant and the parameters τ,σ set to different values to get a set of Schmid filters. The set of nineteen texture features is shown in Figure 3.1. The Schmid filters are rotationally symmetric and the horizontal and vertical Gabor filters help in extracting the response of the given images in those directions.

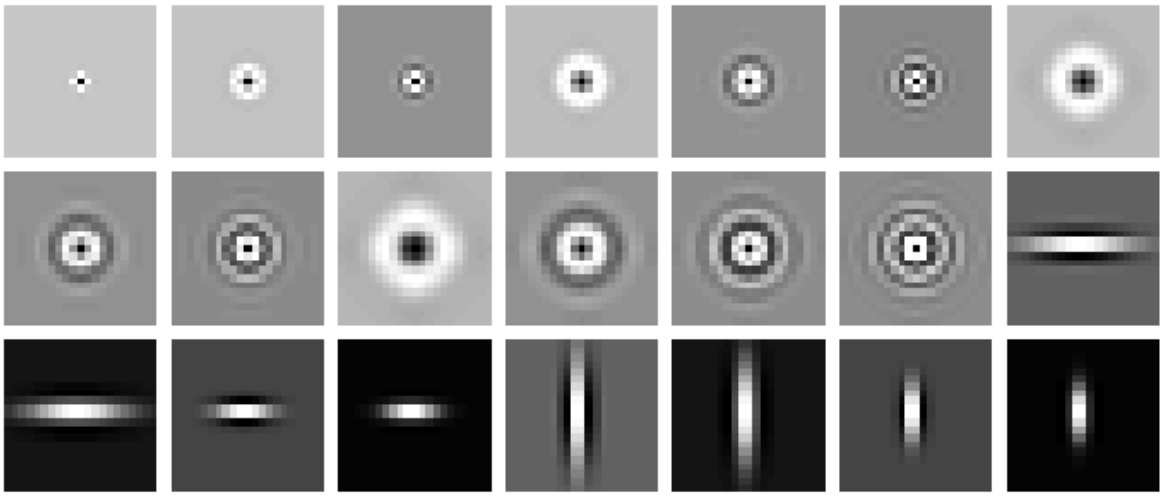

Figure 3.1 19 Texture features (13 Schimd features and 8 Gabor Features)

Here the feature regions can be defined as a set of collection of pixels in the image. Some of the popular feature regions include the simple rectangle, a collection of rectangles [9] or a rectangular shaped region [22]. As the data consists of humans obtained from different horizontal viewpoints and hence the horizontal dimension is not likely to be a contributing factor to the features, therefore as proposed in [2], we select a set of stripes which span the entire horizontal dimension.

To define the range over which the pixel values are counted, we use a feature bin as in [2]. For a histogram, an orthogonal collection of bins is selected to uniformly cover the range of all possible values.

Also, the features computed using this representation include low-level features which are widely used for a variety of existing person re-identification techniques and hence we can consider this representation as generic and representative.

Chapter 4: An Introduction to Partial Least Squares

Partial Least Squares (PLS) is a technique for modeling the relations between sets of observed variables by means of latent variables [42]. PLS assumes that the system which generates observed data or process is driven by a small number of latent variables. PLS gives excellent results when the number of observed variables is much greater than the number of observations and high multi-collinearity (i.e. variables are highly correlated) exists between the variables. The observed data is projected on this latent structure by means of latent variables. PLS creates orthogonal score vectors or latent vectors by maximizing the covariance between different sets of variables.

Now we model the relations between the two blocks of variables. We denote $X \subset R^N$ as an N-dimensional space of variables representing the first block of variables and similarly $Y \subset R^M$ as the second block of variables. PLS models the relations between these two blocks of variables by means of score vectors. Let the observations consist of n data samples from each block of variables. First we center the (n X N) matrix of variables X and the (n X M) matrix of variables Y to have zero mean. Next we decompose X and Y according to the relations:-

$$X = TP^T + E$$
$$Y = UQ^T + F \qquad (4.1)$$

Here T,U are (n X p) matrices of the p extracted score vectors , (N X p) matrix P and the (M X p) matrix Q represent matrices of loading and the (n X N) matrix E and (n X M) matrix F are the matrices of residuals. The PLS finds the weight vectors w,c such that the covariance between X and Y is maximised according to the relation:-

$$[\mathrm{cov}(t,u)]^2 = [\mathrm{cov}(Xw,Yc)]^2 = \max_{|r|=|s|=1}[\mathrm{cov}(Xr,Ys)]^2 \quad (4.2)$$

where $\mathrm{cov}(t,u) = t^T u/n$ denotes the sample covariance between the score vectors t and u. This is very much similar to the NIPALS algorithm [42]. The PLS algorithm further finds the score vectors t and u according to the following iterative procedure until convergence :-

$$\begin{aligned} &1) w = X^T u/(u^T u) \\ &2) \| w \| \to 1 \\ &3) t = Xw \\ &4) c = Y^T t/(t^T t) \\ &5) \| c \| \to 1 \\ &6) u = Yc \end{aligned} \quad (4.3)$$

Furthermore, from [42], it can be deduced that we can also obtain weight vector w according to the first eigenvector of the following eigenvalue problem :-

$$X^T Y Y^T X w = \lambda w \quad (4.4)$$

And finally we can obtain the X and Y space score vectors t and u as in :-

$$t = Xw$$
$$u = Yc \tag{4.5}$$

where the weight vector c is defined as in steps 4 and 5 above. Due to the iterative nature of PLS, after the extraction of the score vectors, the matrices X and Y are deflated by subtracting their rank one approximations based on t and u.

The non-linear PLS method (KPLS) is based on mapping the original input data into a high-dimensional feature space F. In KPLS the weight vectors w,c can not be easily computed since the data is available in the form of inner products. But we can directly compute the score vectors t using the first eigenvector of the following eigenvalue problem obtained from the above two equations.

$$XX^T YY^T t = \lambda t \tag{4.6}$$

The Y-score vectors u can then be easily estimated as :-

$$u = YY^T t \tag{4.7}$$

Now we define a nonlinear transformation of x into a feature space F and denote ϕ as the (n X S) matrix of the mapped X- spaced data $\phi(x)$ into an S-dimensional feature space F. Also define the kernel Gram matrix K as :-

$$K = \phi\phi^T \tag{4.8}$$

where K represents the (n X n) kernel Gram matrix of the cross dot products between all input data points $\{\phi(x_i)\}_{i=1}^{n}$

Along the same lines, we consider a mapping of the second set of variables y into a feature space F_1 and denote by ψ, the (n X S_1) matrix of mapped Y space data $\psi(y)$ into an S_1 dimensional feature Space F1. Again we define the (n X n) kernel Gram matrix K_1 as :-

$$K_1 = \psi\psi^T \tag{4.9}$$

Similarly, the estimates of t and u can be obtained by solving :-

$$KK_1t = \lambda t$$
$$u = K_1t \tag{4.10}$$

Further the mapped data in the feature space F can be centralized using the mapping :-

$$K \leftarrow (I_n - \frac{1}{n}1_n1_n^T)K(I_n - \frac{1}{n}1_n1_n^T) \tag{4.11}$$

where I_n is an n-dimensional identity matrix and 1_n represents the (n X 1) vector with elements equal to 1. Similarly we can center the matrix K1.

After the extraction of the new score vectors t and u, the matrices K and K_1 are deflated by subtracting their rank-one approximations based on t and u.

Finally, the KPLS algorithm can be summarized as:-

1.Randomly Initialize u

$$2.t = \phi\phi^T u \ldots . t \leftarrow t / \| t \|$$
$$3.c = Y^T t \qquad (4.12)$$
$$4.u = Yc \ldots . u \leftarrow u / \| u \|$$

5.Repeat steps 2-5 until convergence

6.Deflate $\phi\phi^T, Y$ matrices: $\phi\phi^T \leftarrow (\phi - tt^T\phi)(\phi - tt^T\phi)^T, Y \leftarrow Y - tt^T Y$

By applying the kernel trick $\phi(x_i)^T \phi(x_j) = K(x_i, x_j)$, it can be seen that $\phi\phi^T$ represents the (n X n) kernel Gram matrix K of the cross dot products between all mapped input data pints $\{\phi(x_i)\}_{i=1}^n$, hence using this kernel function in the 6th step of the above algorithm, we can write the deflation of the K matrix after extraction of the t components as :-

$$K \leftarrow (I - tt^T)K(I - tt^T) = K - tt^T K - Ktt^T + tt^T Ktt^T \qquad (4.13)$$

where I is the n dimensional identity matrix.

Next we use the matrix of regression coefficients to make predictions on the test data samples. The details for matrix B can be found in [42]. It is defined as :-

$$B = \phi^T U (T^T KU)^{-1} T^T Y \qquad (4.14)$$

And to make predictions on training data, we can use the equations :-

$$\hat{Y} = \phi B = KU(T^T KU)^{-1} T^T Y = TT^T Y \qquad (4.15)$$

And for predictions on testing data, the matrix of regression coefficients can be used as follows :-

$$\hat{Y}_t = \phi_t B = K_t U (T^T K U)^{-1} T^T Y \tag{4.16}$$

where ϕ_t is the matrix of mapped testing points and K_t is the (n_t X n) test matrix whose elements are $K_{ij} = K(x_i, x_j)$ where $\{x_i\}_{i=n+1}^{n+n_t}$ and $\{x_j\}_{j=1}^{n}$ are the training and testing points respectively. Furthermore, the test Gram matrix can be centralized in the same way as the train gram matrix by using the following equation :-

$$K_t \leftarrow (K_t - \frac{1}{n} 1_{n_t} 1_n^T K)(I - \frac{1}{n} 1_n 1_n^T) \tag{4.17}$$

where the vectors have the same meaning as in the formulation for the train Gram matrix.

More in-depth details can be found in [42].

Chapter 5: Formulating the Matching Problem

In this chapter, we formulate the algorithm and present a numerical analysis of the same. First, we randomly form the watch-list (the target set) by selecting some images (ten or six in the target set) from the person indices. The remaining images constitute the non-target set. Further, we divide the target and non-target images into training and testing images by randomly selecting from both.

Next, we normalize the images and extract the feature set from each training image. This is done by dividing the image into six horizontal stripes and then extracting a mix of RGB features and other textural features like Gabor and Schimd features and forming a 2784 dimensional feature vector. Now the training set consists of two classes namely the target set and the non-target set. We then use KPLS discussed in Chapter 4 to reduce the number of dimensions for each training image such that the target and non-target images are separated in the new latent space.

So for any new probe image, we first extract its 2784 dimensional feature vector and then transform it to the new latent space. Finally, to classify any probe image, we use a simple Euclidean distance based nearest neighbor classifier. This classifies the probe image into the target image or a non-target image. So in any surveillance setting, we can verify if a probe image belongs to any target image person or not. In addition with the help of human intervention, we can identify the true identity of that probe image. Taking this a step further, after verifying if the image belongs to the target set or not, we can fully automate the process by using a distance-based classifier to check the true identity of the probe image. This way the dependency on the human operator can be reduced. The results are presented using a

rank based histogram which shows the rank within the target images of the given probe image. We also present the correct matching rank of the probe image from the target image set for some randomly chosen probe images.

The above process in which we have six or ten images in the target set is known as multi-shot classification whereas when we have a single image in the target set, we call it the one-shot classification. In one-shot classification, we form classifiers for each and every target image and then for any new probe image we project that image on every classifier and check which classifier gives the closest distance to the target set. This one-shot verification gives a better accuracy but a drawback of this method is that we have to build classifiers for each target image and have to project the probe image on each classifier. In surveillance-based tasks, this increases the overhead which is in many scenarios not acceptable.

The discussion on the selection of the parameters for the classification process, namely, the number of latent variables and the Gamma value for the Radial Basis Function (RBF) kernel is presented in the experiments section. The best results were obtained using the RBF kernel and the results are presented only for the same.

A flowchart depicting the above process is shown in Figure 5.1.

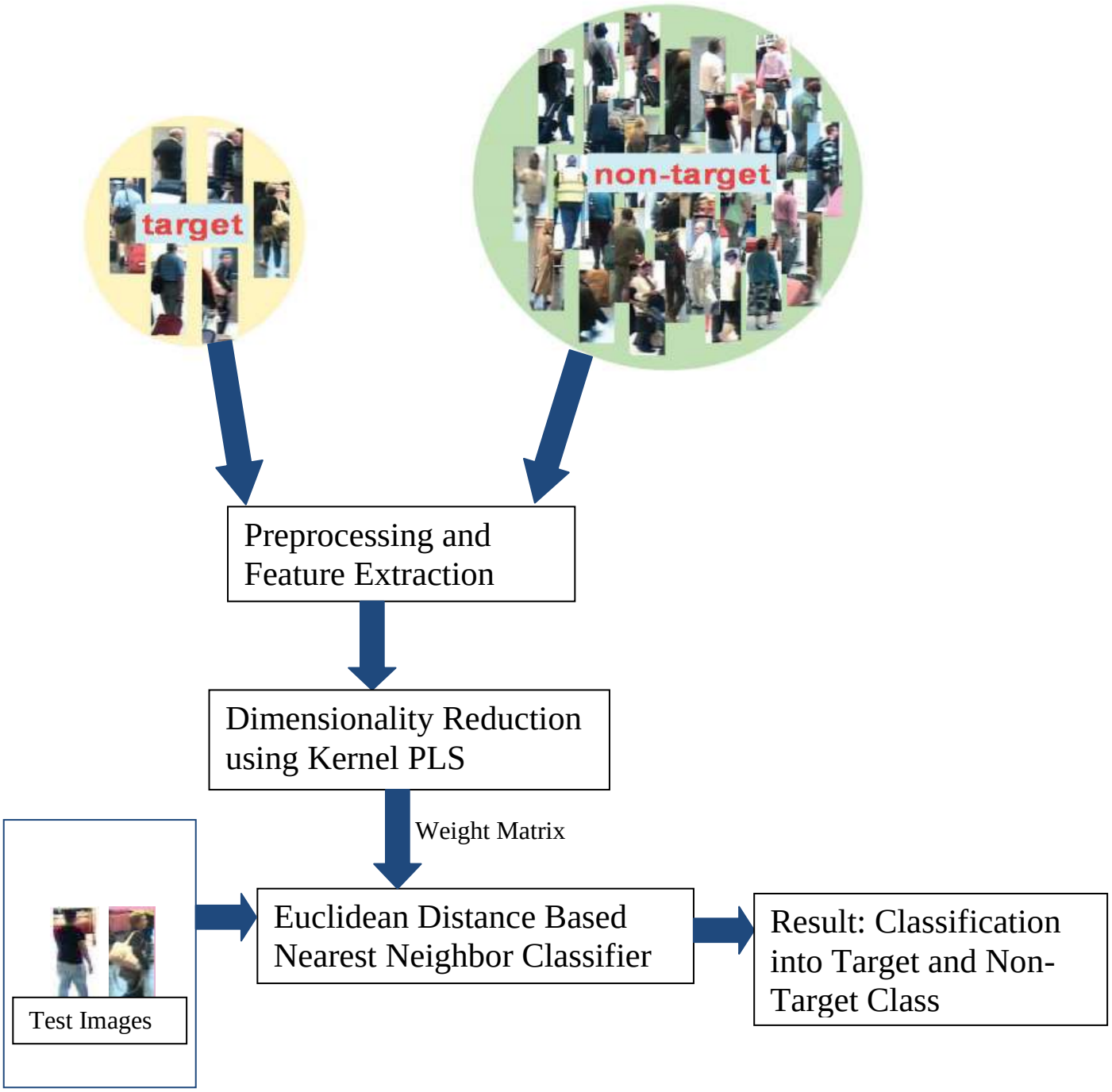

Fig 5.1 Flowchart for the Algorithm

We assume that we are given N_t target training data samples from m_t different target indices $C_t^1, C_t^2 \ldots\ldots\ldots C_t^{N_t}$ denoted by $\{\mathbf{x}_t^i, y_t^i\}_{i=1}^{N_t}$ where $y_t^i \in \{C_t^1, C_t^2 \ldots\ldots\ldots C_t^{N_t}\}$ and $\mathbf{x}_t^i$ denotes the i^{th} training sample. Further, we have unlabeled data from other subjects which form the non-target sample $\mathbf{x}_t^1, \mathbf{x}_t^2 \ldots\ldots\ldots \mathbf{x}_t^{N_{Nt}}$. Here the number of non-target samples is much greater than the number of target samples. Now the problem comprises of coming up with a model which can separate these target samples with the non-target samples. In [1], the authors separate the two sets using a distance metric to define the relevant and irrelevant samples whereas in this thesis, we use KPLS to separate the two classes in the latent variable space. We input the two classes denoted by $\{\mathbf{x}_t^i, y_t^i\}_{i=1}^{N_t}$ and $\mathbf{x}_t^1, \mathbf{x}_t^2 \ldots\ldots\ldots \mathbf{x}_t^{N_{Nt}}$ as the target and non-target samples respectively into the KPLS algorithm. An important point here is to remember that the non-target samples are unlabeled and are considered in neutrality. The final step comprises of using the weight matrix to convert the probe image into the latent space domain and then using a nearest-neighbor Euclidean distance-based classifier for classification into the watch-list.

Chapter 6: Datasets

We use two datasets for evaluation purposes namely the ETHZ dataset [39] and the popular VIPeR Dataset [40].

6.1 ETHZ Dataset

This dataset consists of 146 people with 8555 images in total. Each image was normalized to 128 X 64 pixels. The images of subjects were taken from a moving camera in a busy street. As discussed earlier, the labels of all non-target images used are assumed to be unknown. This dataset has been captured using moving cameras. Such a camera setup provides a range of variations in the appearance of a person which can be judiciously used for training the classifier. Figure 6.1 shows a scene captured from the camera. Several scenes were captured in the same way and the individual person images were extracted.

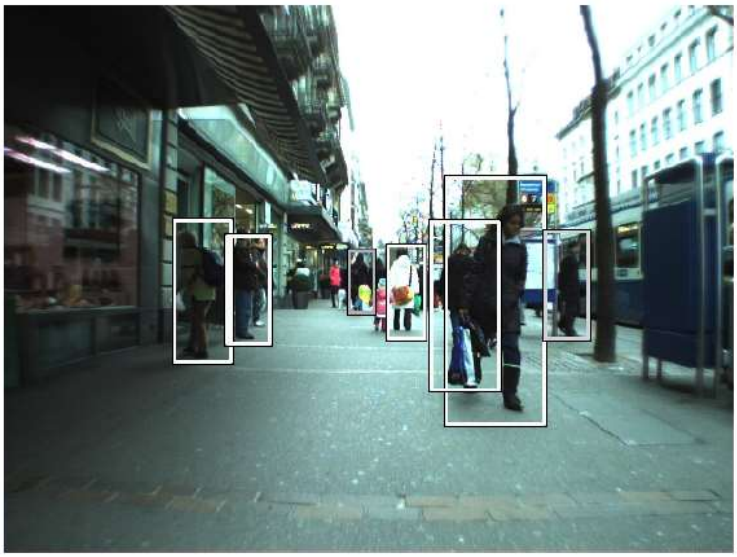

Figure 6.1 A scene from the dataset ETHZ

For the following dataset three samples of video sequences were taken from the cameras. Further the video sequences were split into frames as follows:-

Sequence #1 is composed of 1000 frames with 83 different people.

Sequence #2 is composed of 451 frames with 35 people.

Sequence #3 is composed of 354 frames with 28 people.

Figure 6.2 shows some samples of the same person extracted from different frames. It is clear that changes in pose and illumination due to the moving camera makes the problem more challenging which calls for features which are rotation invariant with different types of texture and color features. The main challenges for this dataset are illumination changes and occlusion on peoples' appearance.

Figure 6.2 Samples of the same person extracted from diffeent frames

To set up the problem using the ETHZ Dataset, first we extract or assign some subjects (six and ten) in the watch-list (target set) whereas the rest form the non-target set whose labels are not known (half indexes for training and rest for testing) . Further, we randomly divide the target samples into training and testing sets.

6.2 VIPeR Datset

The Viewpoint Invariant Pedestrian Recognition (VIPeR) dataset consists of 632 pedestrian image pairs taken from two camera views. In this dataset, the image size of each person was normalized to 128 X 64 pixels. The motivation behind developing this dataset [40] was to construct surveillance scenarios which require the ability to track pedestrians in large open environments such as public places and airport terminals. Figure 6.4 shows some images from the dataset. Here each column is one of the same person example pairs. This dataset also presents a wide range of viewpoint, pose and illumination changes. Here the view angle change is the most significant cause of appearance change with most of the matched image pairs containing a front/back view and one side-view. There is little occlusion with some illumination change.

Figure 6.3 Sample images from VIPeR Dataset

After defining the indices of people to be grouped under target samples, we use one image for training and one for testing. The non-target samples are classified in the same way as in the other dataset.

It can be seen that the two datasets have different characteristics such as outdoor/indoor, variation in view angle, occlusion, pose and illumination thereby making them ideal for our evaluation.

Chapter 7: Experiments, Results and Discussions

The experiments were set up as in [1]. We compare the proposed method using a watch-list with four other methods, namely, Transfer RankSVM, Transfer PRDC, PRDC and RankSVM. The results obtained were compared on different datasets discussed in Chapter 6. The above methods cannot be used on ViPER Dataset since these methods require at least two images in the target samples to form the relevant or irrelevant pairs whereas our method doesn't have any such problems.

For each dataset, we randomly selected all images of p persons (classes) to set up the target dataset and the remaining person indices to form the non-target set. Then, for the target dataset, we randomly divided it into a training set and a testing set, where q images of each person were randomly selected for training. Next, the non-target data was also divided into training and testing sets such that images of half of non-target subjects in the dataset were used as training non-target images and all the rest were selected as testing non-target images so that there is no overlap of subjects between training non-target images and testing non-target images. Also, it is assumed that the labels of non-target subjects are not known and all the non-target images are clubbed together in a single class of non-target people. We present the results for two scenarios where we set the number of target subjects in the watch-list to six and ten i.e. p=6 and p=10. Further, we select two images as training samples for each target person apart from the experiments in ViPER Dataset where we take only one image to form the training target images. Since the number of images in training target data is much less than the training non-target data, we remove this imbalance by replicating the number of images in the target set.

To compute the curve in each case we define 2 quantities as in [1] namely the True Target Recognition Rate which shows how well a true target has been verified and False Target Recognition Rate which shows how bad a false target has passed through the verification tests. The TTR and FTR rates are defined as follows:-

$$\text{TTR} = \frac{\text{query target images that are verified as one of the target people}}{\text{query target images from target people}}$$

$$\text{FTR} = \frac{\text{query non - target images that are verified as one of the target people}}{\text{query non - target images from non - target people}}$$

Finally, we draw the curve for FTR vs. TTR values for each method while changing the threshold value. Also tables comparing the values of the True Target Rate at a specific value of False Target Rate (30% and 50%) are reported for each method. This gives an indication of the efficiency of each of the methods.

Further, in case of multi-shot verification we assume that after verifying if the test image belongs to the watch-list or not, the human operator can verify to which class the test image actually belongs to (out of ten or six subjects in the watch-list). To automate the whole process, we simply use the nearest neighbor rule on the PLS reduced data to find the nearest class to the given sample. This also achieves good levels of accuracy and as future work we can try different classifiers to fully automate the process.

We perform cross validation for deciding the parameters for the RBF Kernel for the KPLS and the dimensions to which the data should be reduced. The plots for the same are shown for the ETHZ Dataset.

For the ETHZ Dataset, we show the results from one- shot verification since it directly performs verification on the person's identity of any query image. But a

disadvantage of one-shot verification is that it does not measure explicitly the probability that the person of the query image is on the watch list.

For a fair comparison, the features extracted in all the cases are similar.

7.1 Experiments on ETHZ Dataset

a. For ten subjects in the watch-list, we compare the multi-shot verification using the ROC curve for the proposed method and other set-based techniques. Figure 7.1 shows the (TTR vs FTR) curve for the same. It can be seen that KPLS performs better than the other transfer and non-transfer techniques. This is because KPLS aims at separating the target and non-target classes directly in the low-dimensional space by maximizing their covariance between the two classes. We also see that KPLS beats PLS since it allows for a non-linear dimensionality reduction. The True target Rate (at FTR = 50%) for Kernel PLS is 92.30% as compared to the best technique i.e. Transfer PRDC which gives a TTR of 85.09% as reported in [1]. In table 7.1, we show the results for a fixed FTR of 30% and 50% from which we conclude that KPLS gives best results. It is also clear that PLS doesn't perform at par with KPLS, which reinforces that, the data in the two target and non-target sets is nonlinear.

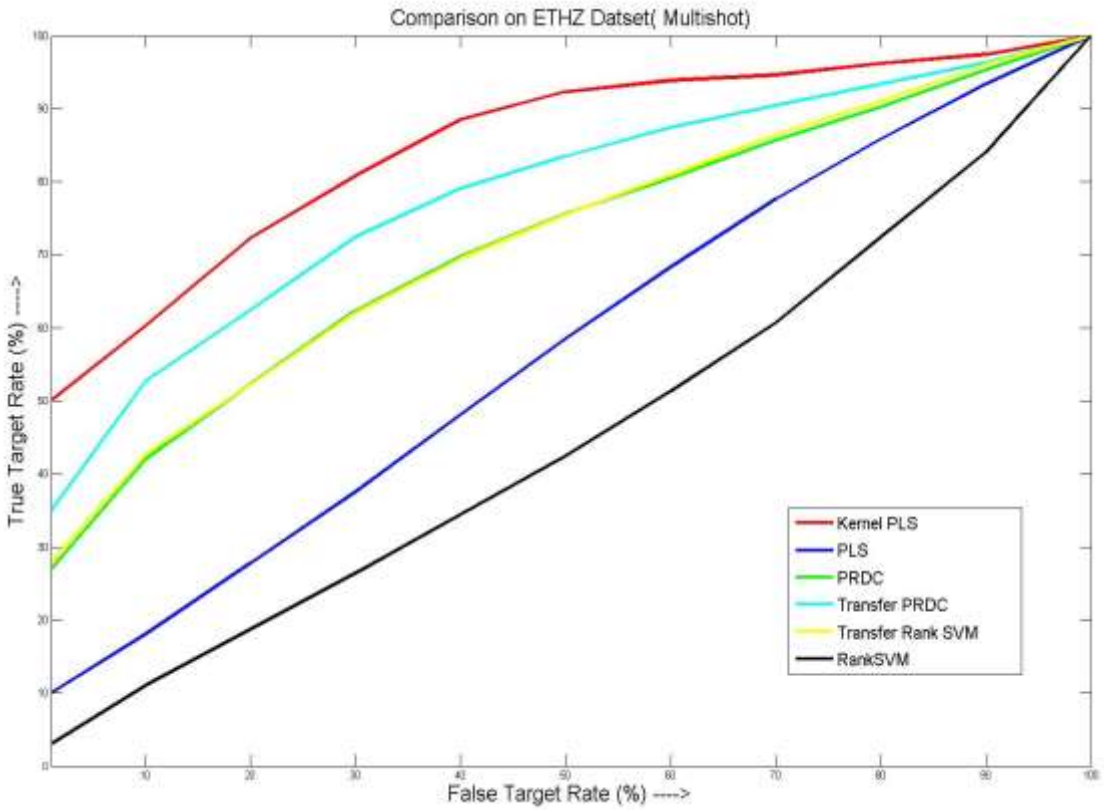

Figure 7.1 Comparison on ETHZ Dataset(Multishot)

METHODS	TTR (at FTR = 30%)	TTR (at FTR = 50%
Kernel PLS	**88.46154**	**92.30769**
PLS	37.4038	58.8462
Rank svm	25.68	41.51
T ranksvm	63.19	77.24
PRDC	64.63	77.12
TPRDC	74.19	85.09

Table 7.1 Comparison on ETHZ Dataset (watch-list = 10)

b. In a similar manner we compare the results obtained from employing six people in the watch-list. Table 7.2 shows KPLS performs the best among considered methods.

METHODS	FTR = 30%	FTR = 50%
Kernel PLS	**85.7143**	**96.2963**
PLS	37.5	61.9048
Rank svm	25.82	45.15
T ranksvm	63.67	75.22
PRDC	64.87	77.69
TPRDC	76.08	85.5

Table 7.2 Comparison on ETHZ Dataset (watch-list = 6)

c. After experimenting with linear, polynomial and RBF kernels, we chose the RBF kernel as it gives the best results. The optimum value of Gamma for the RBF kernel chosen is 4. We use the training data and plot (TTR (at FTR = 30%) vs. Gamma) and (TTR (at FTR = 60%) vs. Gamma), and choose that Gamma for which we get the maximum value for the TTR values. The graphs are shown in Figure 7.2a and Figure 7.2b

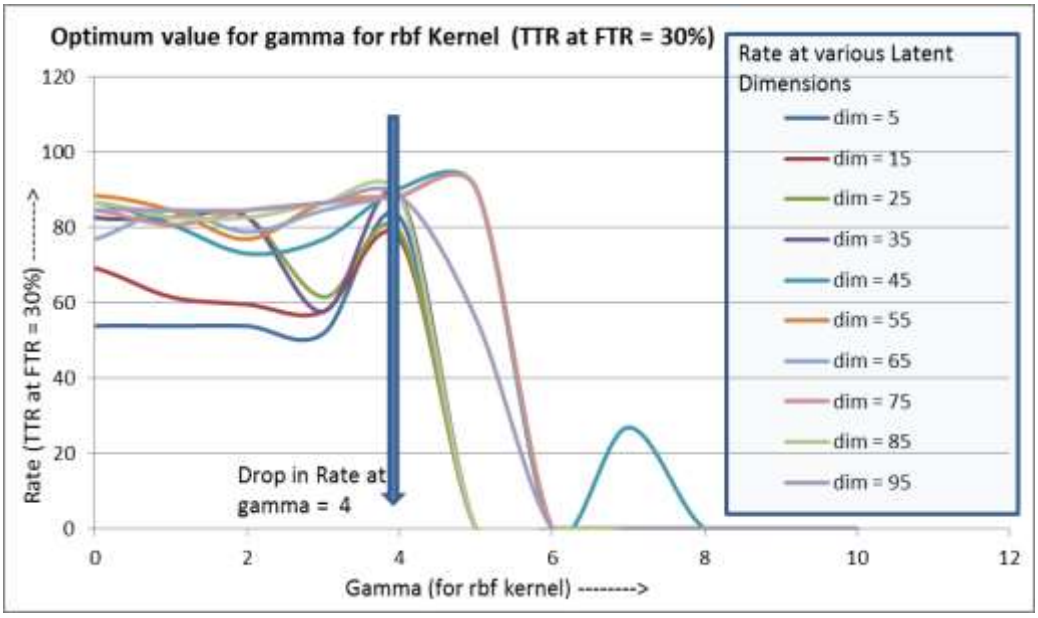

Figure 7.2a Optimum value for Gamma for RBF Kernel (TTR at FTR = 30%)

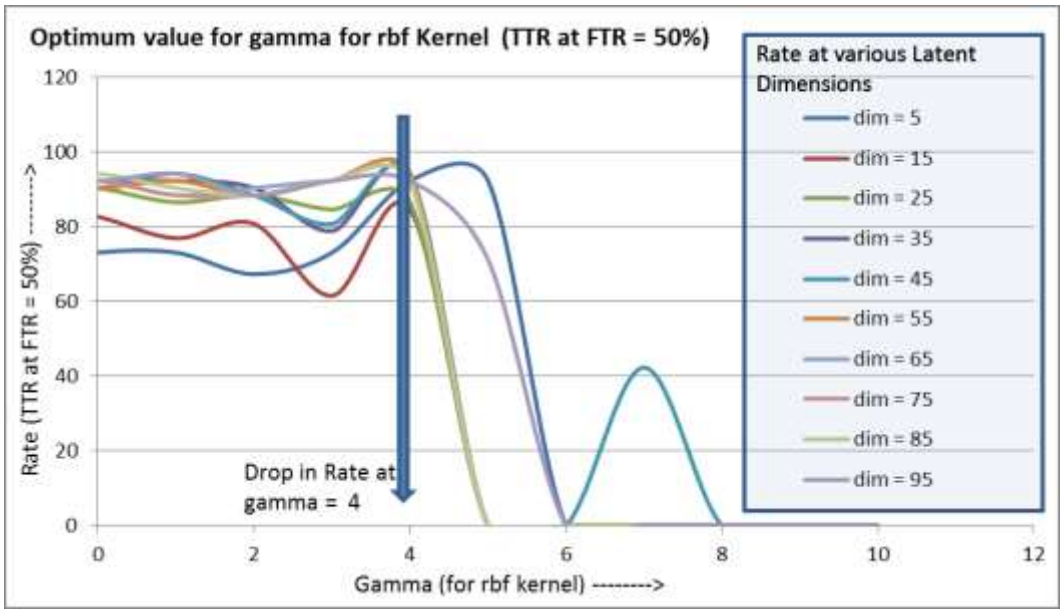

Figure 7.2b Optimum value for Gamma for RBF Kernel (TTR at FTR = 60%)

d. Similarly we obtain the optimum value (= 75) of the number of latent variables to which the data should be reduced to by KPLS. This value is selected by keeping in mind that the rate achieves maximum value and almost becomes constant after the optimum value of 75 in both the figures. The graphs are shown in Figure 7.3a and Figure 7.3b respectively.

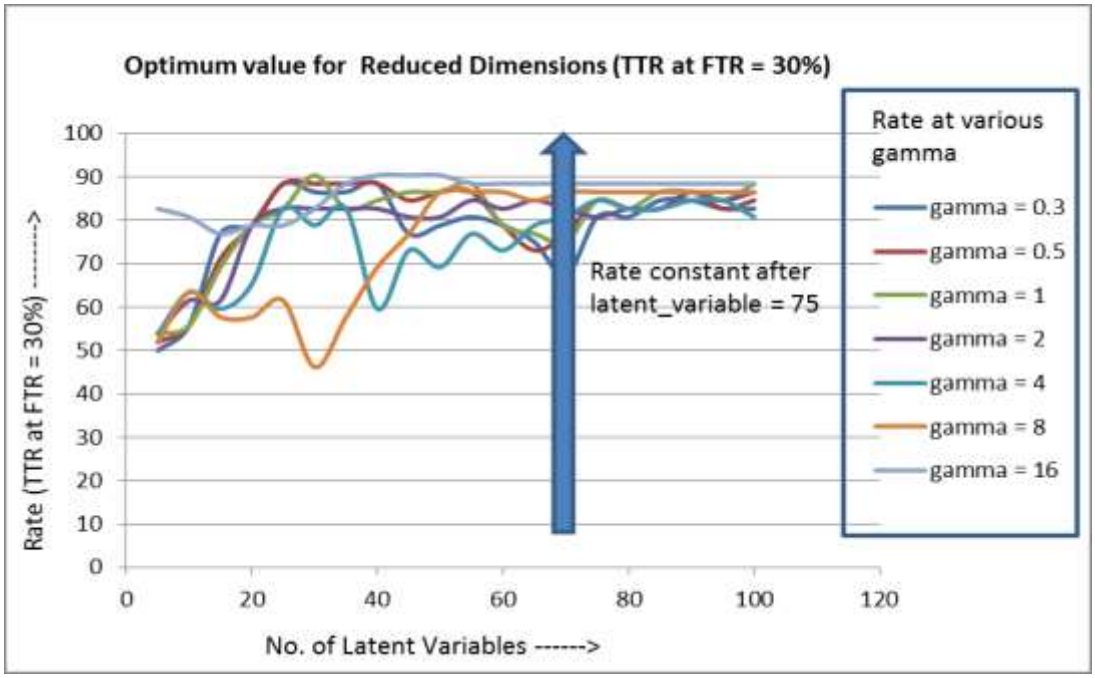

Figure 7.3a Optimum value for Reduced Dimension (TTR at FTR = 30%)

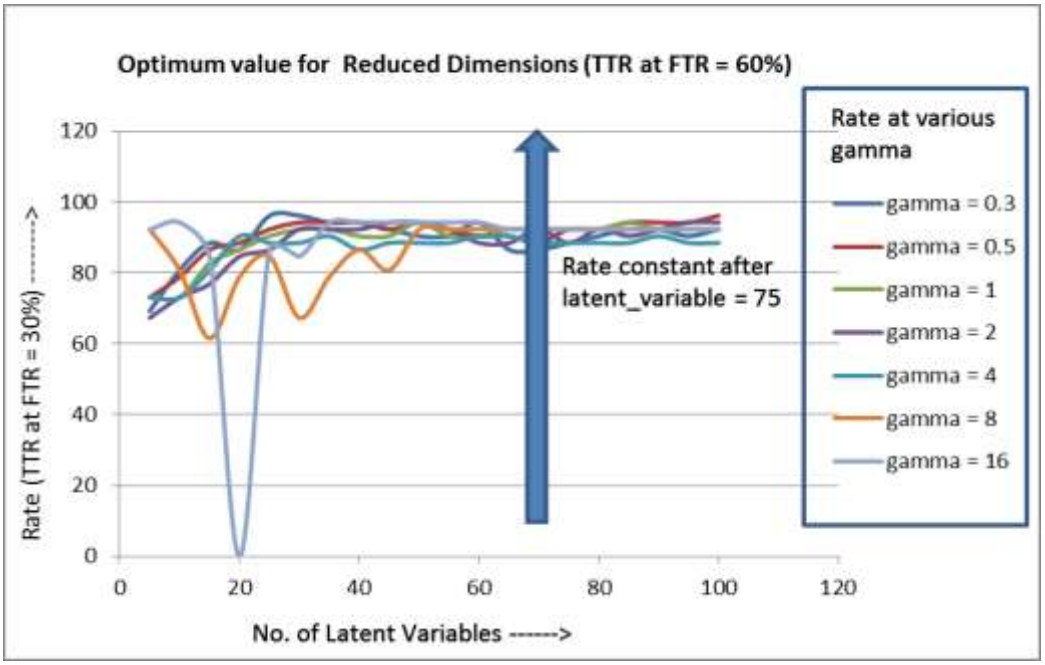

Figure 7.3b Optimum value for Reduced Dimension (TTR at FTR = 60%)

e. After verifying if the probe image belongs to the target or not, a human can easily classify to which of the target classes the image belongs. To further fully automate the process after verifying whether the given test image is from the watch-list or not, we directly use an Euclidean distance-based classifier to classify in which out of the ten classes in the watch-list, the probe image belongs. The Bargraph in Figure 7.4 shows the percentage of test images classified correctly on the basis of ranking (out of ten) i.e. the ranked matching rate. This is similar to the CMC (Cumulative matching Characteristic) curve. This shows that the KPLS method can be easily automated. In Figure 7.5, we also show the probe (test) image and the ten gallery images and their respective correct matches for some examples from the ETHZ Dataset. We can see in the first example why the probe image

couldn't be identified correctly since the true match and the false match (ranked 1) look almost the same in appearance which makes the recognition process very difficult. We have shown only the first eight matches(out of ten) for three random images. The correct match is shown by red square. It can be seen that within the first three matches out of the target person images, a full accuracy of 100% is achieved which shows that all the probe images have been classified correctly from the target set.

Figure 7.4 Ranked Matching Rate (for ETHZ Dataset)

Figure 7.5 Examples of Matching from ETHZ Dataset

f. One-shot Verification :- Here we use KPLS for building the classifier for each person in the target list. This type of verification is expensive but gives better results since the classifier brings out discriminating features from the target sample. Again we compare the results with the ones mentioned in [1] using RankSVM and Transfer RankSVM. Table 7.3 shows the True Target rate for a fixed False Target Rate and Figure 7.6 shows the corresponding ROC curves.

METHODS	FTR = 30%	FTR = 50%
Kernel PLS	**91.3043**	**95.6522**
PLS	67.3913	78.2609
Rank svm	3.7	6.3
T ranksvm	92.33	96.05
PRDC	88.52	93.8
TPRDC	87.16	93.78

Table 7.3 Comparison on ETHZ Dataset (one-shot)

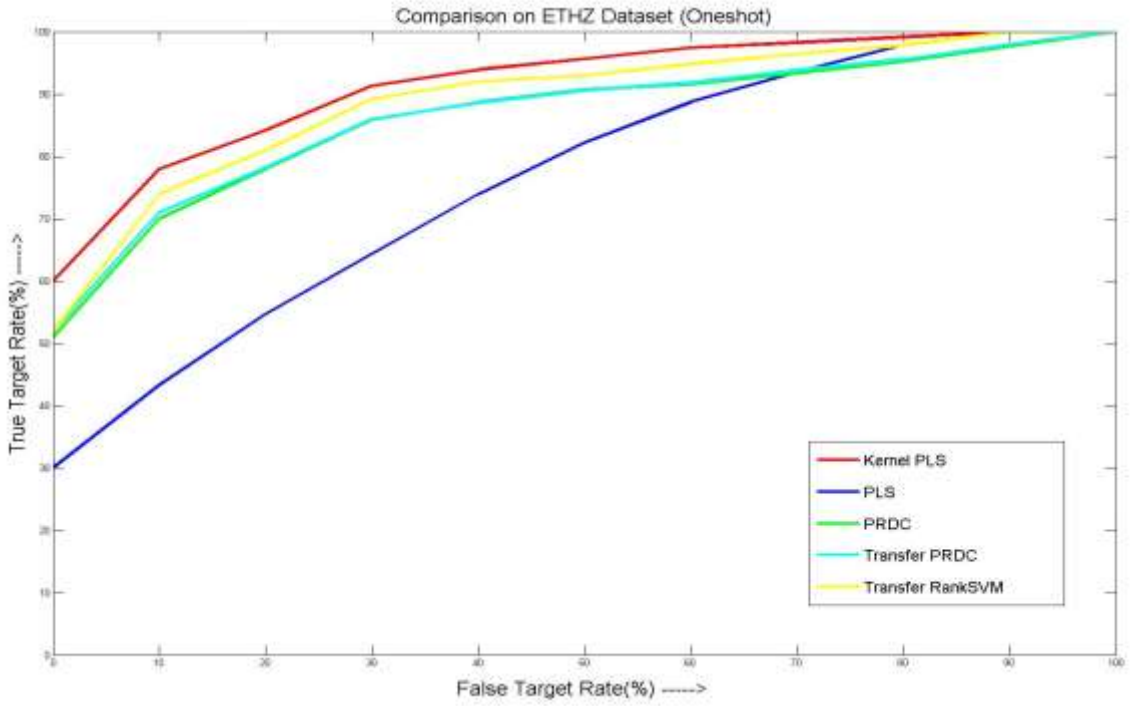

Figure 7.6. Comparison on ETHZ Dataset (One Shot)

7.2 Experiments on ViPER Dataset

For ten persons in the watch-list, we compare the multi-shot verification algorithm using the ROC curve for the proposed method with the Euclidean Distance-based classifier. Figure 7.10 shows the ROC curve for the same. It can be seen that KPLS performs better than the other techniques as was the case for the other two datasets. Here the transfer Rank SVM and other similar techniques cannot be used since they require at-least two images for each class in the target samples which is not possible in the case of ViPER Dataset.

Here we chose the Gamma for the RBF Kernel as 32 and the number of latent variables for Kernel PLS as 70. These values were obtained using cross validation from the training samples.

In table 7.6, we show the results for a fixed FTR of 30% and 50% which shows the TTR rate to be the best when KPLS is used.

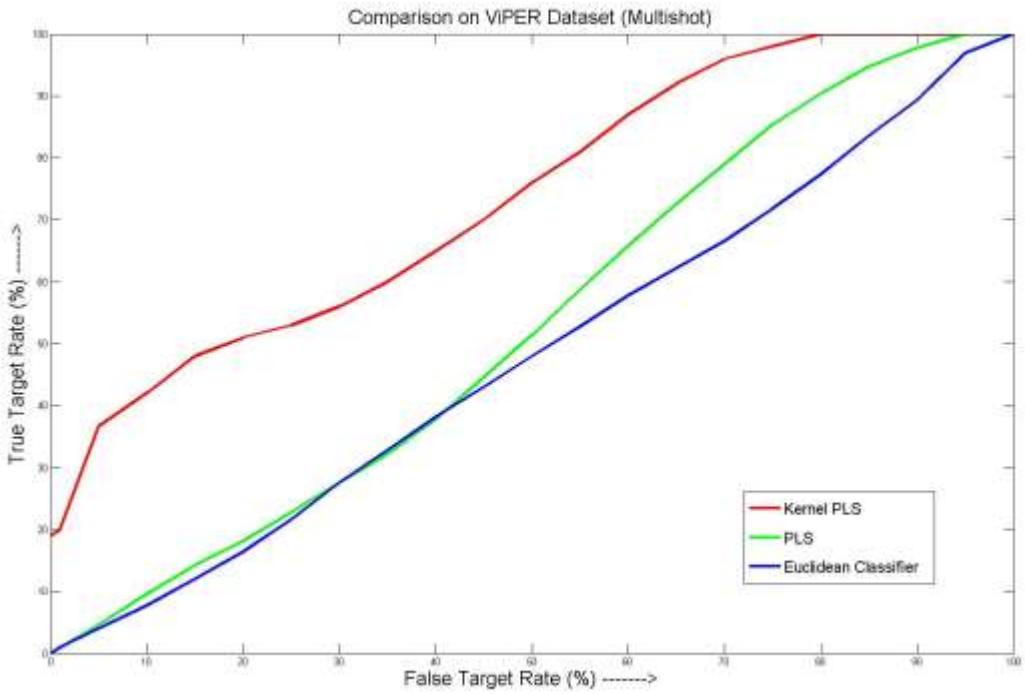

Figure 7.7 Comparison on ViPER Dataset (Multi-shot)

METHODS	FTR = 30%	FTR = 50%
Kernel PLS	**50**	**70**
PLS	26	51
Euclidean Classifier	26	49

Table 7.4 Comparison on ViPER Dataset (watch-list = 10)

a. In a similar manner we compare the results obtained from employing six subjects in the watch-list. Table 7.7 shows the results where KPLS performs the best among the compared methods.

METHODS	FTR = 30%	FTR = 50%
Kernel PLS	**33.33**	**83.33**
PLS	38.88	61.11
Euclidean Classifier	33.33	50

Table 7.5 Comparison on ViPER Dataset (watch-list = 6)

From the extensive experiments performed on the two Datasets, we can conclude that the proposed method performs better than the baselines.

Chapter 8: Conclusion and Future Work

In this thesis, a framework for a watch-list based surveillance system by largely focusing on the problem of person re-identification was presented. In terms of person re-identification a set-based classification technique was presented which aims to maximize the separation between the target and the non-target sets by projecting them onto a set of latent vectors. This set-based approach gives an advantage over building individual classifiers as it exploits the inter and intra class distance between the samples very effectively. By using KPLS, the high dimensionality of the data is reduced thereby making the problem manageable. And finally a simple nearest-neighbor classifier is employed which gives high levels of accuracy. Also a human operator can identify the correct match out of the target samples with ease. This classification is further aided by a careful selection of features comprising of both color and textual features which are rotation invariant. After classifying the probe image into the target set, we further try to fully automate the process by removing the human dependency with the use of a simple Euclidean distance classifier which gives decent rank matching results.

There has been limited work on set-based classification which constructs a single classifier for the whole set and the proposed method outperforms the one introduced in [1] which is based on the ranking of relevant and irrelevant pairs. This claim is supported by the experiments performed on two datasets namely ETHZ and the ViPER Dataset. The method proposed is faster than [1], i.e. once we find the mapping using the training data, classification of any new probe image is very quick whereas in assigning ranking, it's very time consuming to build the relevant and

irrelevant samples for any new probe image as the dataset is huge. Secondly as seen through the experiments and results, the accuracy achieved by the proposed method is higher than any of the methods proposed in [1].

In the proposed algorithm, we have used the Radial Basis Functions for Kernel Partial Least Squares. Since the choice of kernel is left to us, we can further get optimal performance by learning the kernel. By supplying the data to the learning algorithm, the algorithm can discover the best kernel for the problem at hand. It involves feature selection, feature weighing, distance and transfer learning techniques followed by solving the optimization problem. Secondly, one of the drawbacks of this algorithm or for that matter any person re-identification framework is that it assumes that the appearance of the person doesn't change. In such a case we need better features for face and gait recognition to address the changes in appearance. This type of classification on sets can also be extended for other biometric including iris, face, fingerprint recognition etc. Furthermore, boosting can be experimented with to increase the accuracy of classifiers like SVMs.

Bibliography

[1] W.S. Zheng, S.Gong and T. Xiang, “Transfer Re-identification: From Person to set base Verification” in IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[2] D. Gray and H. Tao, “Viewpoint invariant pedestrian Recognition with an ensemble of localized features” in European Conference on Computer Vision, 2008.

[3] U. Park, A. Jain, I. Kitahara, K. Kogure, and N. Hagita, “Vise, Visual search engine using multiple networked cameras” in International Conference on Pattern Recognition, 2006.

[4] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu, “Shape and appearance context modeling” in IEEE International Conference on Computer Vision, 2007

[5] W. Hu, M. Hu, X. Zhou, J. Lou, T. Tan, and S. Maybank, “Principal axis-based correspondence between multiple cameras for people tracking in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol,28(4): 663–671, 2006

[6] M. Farenzena, L. Bazzani, A. Perina, M. Cristani, and V. Murino, “Person re-identification by symmetry driven accumulation of local features in IEEE Conference on Computer Vision and Pattern Recognition, 2010

[7] W.-S. Zheng, S. Gong, and T. Xiang. “Person re-identification by probabilistic relative distance comparison” in IEEE Conference on Computer Vision and Pattern Recognition, 2011

[8] N. Gheissari, T. Sebastian, and R. Hartley, “Person re-identification using spatiotemporal appearance” in IEEE Conference on Computer Vision and Pattern Recognition, 2006.

[9] P. Dollar, Z. Tu, H. Tao, and S. Belongie, “Feature mining for image classification” in IEEE Conference on Computer Vision and Pattern Recognition, 2007

[10] S. Bak, E. Corvee, F. Br´ emond, and M. Thonnat, “Person re-identification using spatial covariance regions of human body parts” in IEEE International Conference on Advanced Video and Signal Based Surveillance, 2010.

[11] D. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino, ”Custom pictorial structures for re-identification” in British Machine Vision Conference, 2011
[12] K. Chen, C. Lai, Y. Hung, and C. Chen, “An adaptive learning method for target tracking across multiple cameras” in IEEE Conference on Computer Vision and Pattern Recognition, 2008
[13] O. Javed, K. Shafique, Z. Rasheed, and M. Shah, “Modeling inter-camera space-time and appearance relationships for tracking across non-overlapping views” in Computer Vision and Image Understanding, vol 109(2):146–162, 2008
[14] B. Prosser, S. Gong, and T. Xiang, “Multi-camera matching under illumination change over time” in ECCV Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications”, 2008
[15] A. Gilbert and R. Bowden, “Incremental, scalable tracking of objects inter camera” in Computer Vision and Image Understanding, vol 111(1):43–58, 2008
[16] G. Lian, J. Lai, and W.-S. Zheng, “Spatial-temporal consistent labeling of tracked pedestrians across non-overlapping camera views”in Pattern Recognition, vol 44(5):1121–1136, 2011
[17] C. Loy, T. Xiang, and S. Gong, “Time-delayed correlation analysis for multi-camera activity understanding” in International Journal of Computer Vision, vol 90(1):106–129, 2010
[18] D. Makris, T. Ellis, and J. Black, “Bridging the gaps between cameras “ in IEEE Conference on Computer Vision and Pattern Recognition, 2004
[19] W.S. Zheng, S.Gong and T. Xiang , “Reidentification by Relative Distance Comparison”, in Pattern Analysis and Machine Intelligence, 2012
[20] B. Prosser, W.-S. Zheng, S. Gong, and T. Xiang, “Person re-identification by support vector ranking” in British Machine Vision Conference, 2010
[21] Guo, Y., Shan, Y., Sawhney, H., Kumar, “PEET: Prototype Embedding and Embedding Transition for Matching Vehicles over Disparate Viewpoints” in Computer Vision and Pattern Recognition. IEEE Computer Society Conference on (2007)
[22] Wang, X., Doretto, G., Sebastian, T., Rittscher, J., Tu, P, “Shape and appearance context modeling, in IEEE International Conference on Computer Vision, 2007

[23] Gandhi, T., Trivedi, M, “Person tracking and reidentification: Introducing Panoramic Appearance Map (PAM) for feature representation” in Machine Vision and Applications 2007
[24] Comaniciu, D., Ramesh, V., Meer, “Real-time tracking of non-rigid objects using mean shift” in IEEE Conference on Computer Vision and Pattern Recognition 2000
[25] Dalai, N., Triggs, B., Rhone-Alps, I., Montbonnot, “Histograms of oriented gradients for human detection” in IEEE Conference on Computer Vision and Pattern Recognition 2005
[26] Huang, J., Ravi Kumar, S., Mitra, M., Zhu, W., Zabih, “Spatial Color Indexing and Applications” in International Journal of Computer Vision, 1999
[27] Birchfield, S., Rangarajan, “Spatiograms versus Histograms for Region-Based Tracking” in IEEE Conference on Computer Vision and Pattern Recognition2005
[28] Hu, W., Hu, M., Zhou, X., Lou, “Principal Axis-Based Correspondence between Multiple Cameras for People Tracking” in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006
[29] Hadjidemetriou, E., Grossberg, M., Nayar, “Spatial information in multiresolution histograms” in IEEE Conference on Computer Vision and Pattern Recognition 2001
[30] A. Bosch, A. Zisserman, and X. Muoz, “Image Classification using Random Forests and Ferns in International Conference on Computer Vision, 2007
[31] H. Zhang, A. Berg, M. Maire, and J. Malik, “SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition in IEEE Conference on Computer Vision and Pattern Recognition, 2006
[32] Y. Amit, D. Geman, and K. Wilder, ”Joint Induction of Shape Features and Tree Classifiers” in IEEE Transaction on Pattern Analysis and Machine Intelligence, 1997
[33] Z. Lin and L. S. Davis, “Learning Pairwise Dissimilarity Profiles for Appearance Recognition in Visual Surveillance” in International Symposium on Advances in Visual Computing, pages 23–34, 2008
[34] W. R. Schwartz and L.Davis, “Learning Discriminative Appearance based models using Partial Least squares” in Sibgraph, 2009

[35] R. Herbrich, T. Graepel, and K. Obermayer, “Large margin rank boundaries for ordinal regression” in Advances in Neural Information Processing Systems, 1999
[36] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, “An efficient boosting algorithm for combining preferences” in Journal of Machine Learning Research, vol (4): pg 933–969, 2003
[37] Schmid, C.: Constructing models for content-based image retrieval. Computer Vision and Pattern Recognition. IEEE Computer Society Conference on 2 (2001)
[38] Fogel, I., Sagi, D.: Gabor filters as texture discriminator. Biological Cybernetics 61(2) (1989) 103–113
[39] A. Ess, B. Leibe, and L. Van Gool, “Depth and appearance for mobile scene analysis” in IEEE International Conference on Computer Vision, 2007
[40] D. Gray, S. Brennan, and H. Tao, “Evaluating appearance models for recognition, reacquisition, and tracking” in IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, 2007
[41] R.Rosipal and N.Kramer, “Overview and recent advances in Partial Least squares”, Springer, 2006
[42] R. Rosipal and L.J. Trejo, “Kernel Partial Least Squares Regression in Reproducing Kernel Hilbert Space”, Journal of Machine Learning Research, 2001