UNIVERSITY OF OTTAWA

MASTER DEGREE THESIS

Person re-identification based on and kernel local fisher discriminant analysis and Mahanalobis distance learning

Author:
Qiangsen He

Supervisor:
Robert Laganiere

A thesis submitted in fulfillment of the requirements
for the degree of Master of Applied Science

in the

VIVA lab

School of Electrical Engineering and Computer Science

January 19, 2017

Declaration of Authorship

I, Qiangsen He, declare that this thesis titled, “Person re-identification based on and kernel local fisher discriminant analysis and Mahanalobis distance learning” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

University of Ottawa

Abstract

Faculty of Engineering
School of Electrical Engineering and Computer Science

Master of Applied Science

Person re-identification based on and kernel local fisher discriminant analysis and Mahanalobis distance learning

by Qiangsen He

Person re-identificaiton has become an intense research area in recent years. The main goal of this topic is to check if the individual appeared in other cameras is the same as the one in current cameras. This task is challenging for the variation of illumination, camera angles, the pedestrians' clothes and object sheltering. It's very important to choose robust descriptors and metric learning to improve accuracy. Mahanalobis based metric learning is a popular method to measure similarity. However, since directly extracted descriptors usually have high dimension, it's intractable to learn a high dimensional Mahanalobis matrix. Dimension reduction are used to project high dimensional descriptors to lower dimension space while preserving those discriminative information as much as possible. In this paper the kernel LFDA [35] is used to reduce dimen- sion given that kernelization method can greatly improve re-identification performance for nonlinearity. Then a metric matrix is learned on lower dimensional descriptors based on the limitation that the within class distance is at least 1 unit smaller than the minimum inter class distance. This method turns to have excellent performance compared with other adcanced metric learning.

Acknowledgements

The acknowledgments and the people to thank go here, don’t forget to include your project advisor…

Contents

List of Figures

List of Tables

Chapter 1

Introduction

People re-identification (Re-ID) has been an intense research topic in recent years, whose main goal is to match a given person with those persons with known labels. Person Re-ID has great potential in video surveillance, target detection and tracking and forensic search. However, it is quite challenging since the accuracy is much influenced by many factors like occlusion, illumination variation, camera settings and color response. In Re-ID, those images with known labels are called gallery images and the image used to know its label is called probe image. The probe image and gallery images can be from the same or different camera views, so the viewpoint and illumination between probe and gallery image can be quite different. Also for the different color response of different cameras, the shots of the same person may look different in different cameras. Besides, occlusions between camera and target person can also bring about quite much difficulty. In a word, images of the same person may look different while images of different persons may kook quite the same.

Given a sequence or video of individuals, there are three steps to match person. A simple work flow is shown in figure 1. However, since most of used Re-ID datasets are well copped manually or by a automatic detector, so most Re-ID work will only focus on robust descriptors designing and efficient matching algorithm designing aimed at those well cropped images.

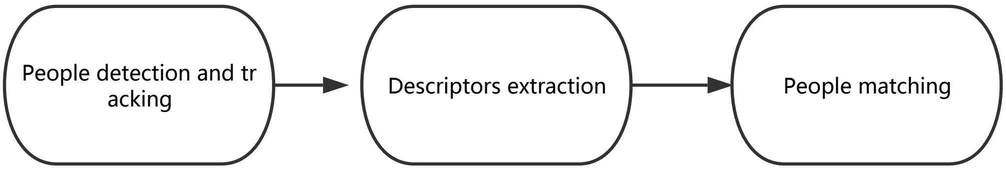


FIGURE 1.1: Re-ID work flow

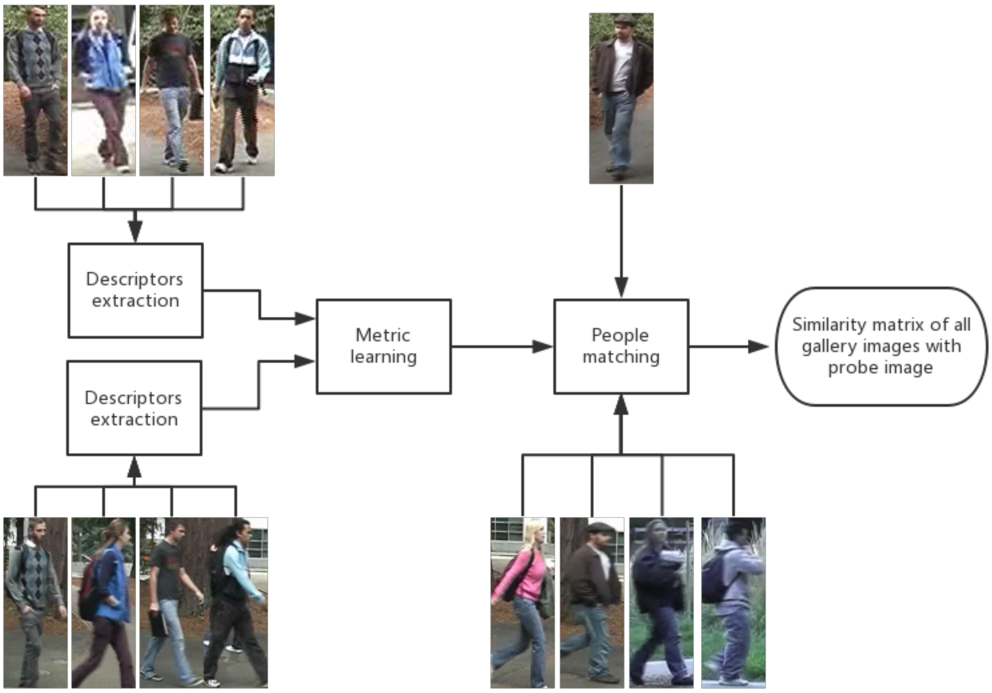


FIGURE 1.2: A typical single shot Re-ID work flow

The first task in Re-ID is to design a robust descriptor to represent images. The descriptor is supposed to contain the key information for each captured person. Basically, the descriptors are supposed to be robust and discriminative. One straightforward way is to extract the color, textural information of images, then the descriptors are used to compute the similarity score. But this method turns out to be not robust caused by illumination variation and camera color response difference and camera angle settings. Therefore, many other advanced descriptors takes into account the correlation of color, texture and position together to improve performance.

The second one is to design the similarity computing methodology. That is, the way to compare how similar two descriptors are. Previous methods use Euclidean distance, Bhattacharyya distance and Mahalanobis distance. The Euclidean distance is the easiest to match descriptors like color and texture descriptors. However, though it's straightforward and easy, it's hard to use Euclidean distance to discriminate images. Many creative metric learning methods have been proposed to compute descriptor similarity. Among them the Mahanalobis distance based metric is most popular. In this method a semi-positive defined matrix M is learned while meeting certain intraclass and interclass distance limitations. Besides, linear discriminant analysis [30] learns a subspace to minimize the within class scatter matrix while maximized inter class scatter matrix. In [43] the null space is proposed that make descriptors of same class collapse into a single point while descriptors of different classes are projected to different points.

1.1 Basic concepts

People re-identification can be divided into a few categories according to different conditions. Some general concepts are listed below.

Open set and close set Re-ID [3] According gallery size and if the gallery size evolves, Re-ID can be divided in to open set Re-ID and close set Re-ID. In close set Re-ID, no new identities will be added to gallery set and gallery size remains the same as time goes by. Besides, the probe set will be a subset of gallery set, that means, the number of unique identities in gallery set will be equal or greater than probe set. In open set Re-ID, the gallery set will evolve as time goes by. Each time a probe image is inputed to the recognition system, the system will judge if it has a corresponding match in the gallery set. If the probe image doesn't match any of the gallery images, it will be regarded as a new identity and will be added to the gallery set. Besides, the probe set is not necessarily the subset of gallery set.

Long term and short term Re-ID According to the time interval between gallery and probe images, Re-ID can be divided into long term and short term Re-ID. In short term means the time interval between gallery and probe images are small, say a few minutes or several hours. In contrary, the long term Re-ID refers to the case that the time interval between gallery and probe images are a few days or even longer. The difference brought by long time interval between gallery and probe images is the variation of individuals' clothes and appearance. If the gallery images are shot a few days ago, the same individual may have changes his suits or take off his bag, then the appearance may change a lot. In this case, it will be much more difficult to recognize the same identity in long term Re-ID. Generally, in most cases we use the short term Re-ID, which guarantees the appearance of same person will remain the same and we only need to consider the difference brought by other factors like viewpoint variation and occlusions.

Single shot and multi shot Re-ID According to the size of sample set for each person, Re-ID can be divided into single shot and multi shots approaches. In single shot case, only one image is provided for a person in a camera view. Single shot Re-ID is challenging because only limited information can be extracted. One example is the VIPeR dataset figure 1.1, in this dataset, for each person only one image is provides in each camera view and the viewpoint of each view is different. In multi shots Re-ID a sequence of images are provided for a person in a camera view. Compared with single shot case, more extra information, like temporal-spatial can be extracted from the sample set. One case of multi-shot dataset is the prid_2011 dataset which provides a long sequence for each person in a single camera view.

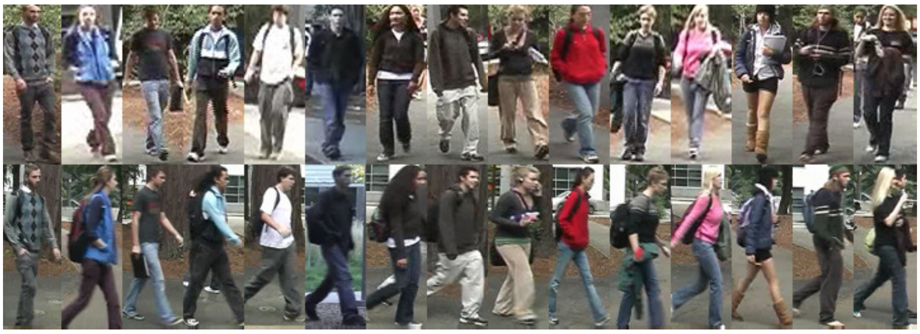


FIGURE 1.3: The VIPeR dataset

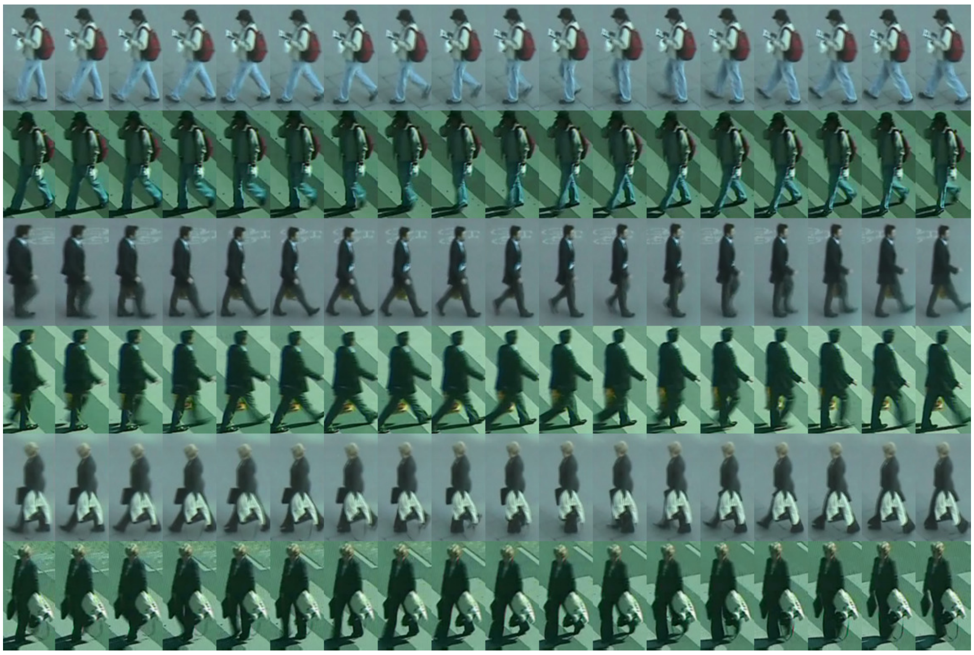


FIGURE 1.4: Samples from prid_2011 dataset

.

1.2 Challenges

Detection, tracking and dataset labelling for supervised learning Though classical person re-identification focus on descriptors designing and matching designing, in real-time application the detection and tracking has to be operated on video frames to get well cropped bounding box images. A good detection and tracking algorithm is necessary for Re-ID. Besides, training the matching algorithm is supervised process, thus we have to know the labels for those training data. Manual labelling turns to be unrealistic for large size dataset. So it's vital to design a automatic labelling algorithm.

Descriptors designing Good descriptors should be robust to people pose variation, outer environment changes and camera settings. Though there have been many kinds of descriptors based on different property like color and texture, it's hard to judge which property is a universally useful for different camera settings. In fact, the robustness, reliability and feasibility depends quite much on different camera settings. What's more, the pedestrian background may bring about much error to descriptors, so it's important to quantify the impact of noisy background. Many works have tried to use segmented foreground of pedestrians, so it's important to design segmentation algorithms. The automatic foreground segmentation for single frame is quite tough since there isn't that much available information compared with video background segmentation. Take VIPeR dataset as an example, there is only one frame for each view of a certain person, thus the segmented foreground masks are imperfect and chance is high that important body parts are lost. A segmented foreground provided by [12] is shown is figure 1.2.

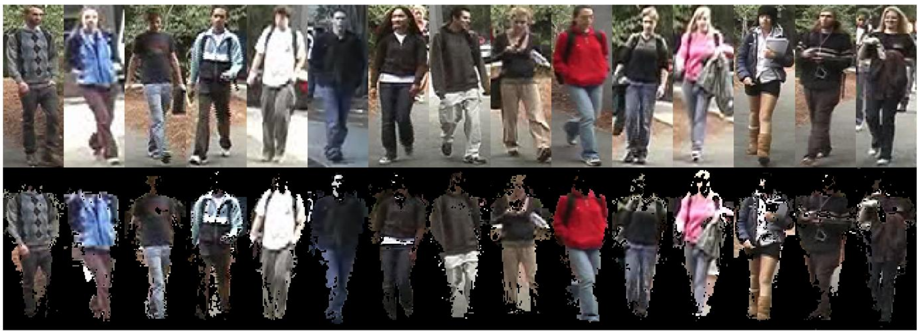


FIGURE 1.5: VIPeR foreground

Efficient matching algorithm designing When designing machine learning algorithms to match persons, there are many limitations. One of them is the small sample size problem [43]. The extracted descriptors usually has a high dimension d but only a small number of sample $n(n << d)$ size are available, underfitting may appear for insufficient data samples with high dimension. Besides, it's also necessary to have a good consideration of intra and inter distance of samples. The intra distance means the distance of two samples with the same class label, while inter class distance is the distance of samples with different class labels.

Feasibility, Complexity and Scalability When applying those descriptors and matching algorithms, we have to consider the its real-time performance. The Re-ID datasets usually has small sample size but in surveillance network much more pedestrians in different cameras can be presented simultaneously. A system like this has plenty of individuals to re-identify, which requires the process time for single probe should be short for low latency. Besides, the since the gallery in this system evolves, it's crucial to design a evolution algorithm for gallery images, that is, how to judge if a person appeared in current camera is a new person to all those gallery images.

1.3 Proposed work

In many previous work, the kernel local fisher discriminant analysis is used as a subspace learning method, and Euclidean distance is usually used in the subspace to measure similarity. In this thesis, the KLFDA [35] method is used a dimension reducing method to project high dimensional descriptors to a lower dimension space.

Compared with other dimension reduction methods, KLFDA is a supervised method and it takes consideration of those intra and inter class information, therefore, much less information are lost after dimension reduction. Then a Mahanalobis distance based matrix M is learned based on the limitation that the distance of people from same class should be at least 1 unit smaller than the distance of people from different classes. A target function that penalizes large intra-class distance and small inter-class distance is created, by iterative computation, when the target function converges the matrix M is thought to be optimal. It turns out that this metric learning have advance performance when compared with other metric learning methods. A workflow of proposed work is in figure 1.6.

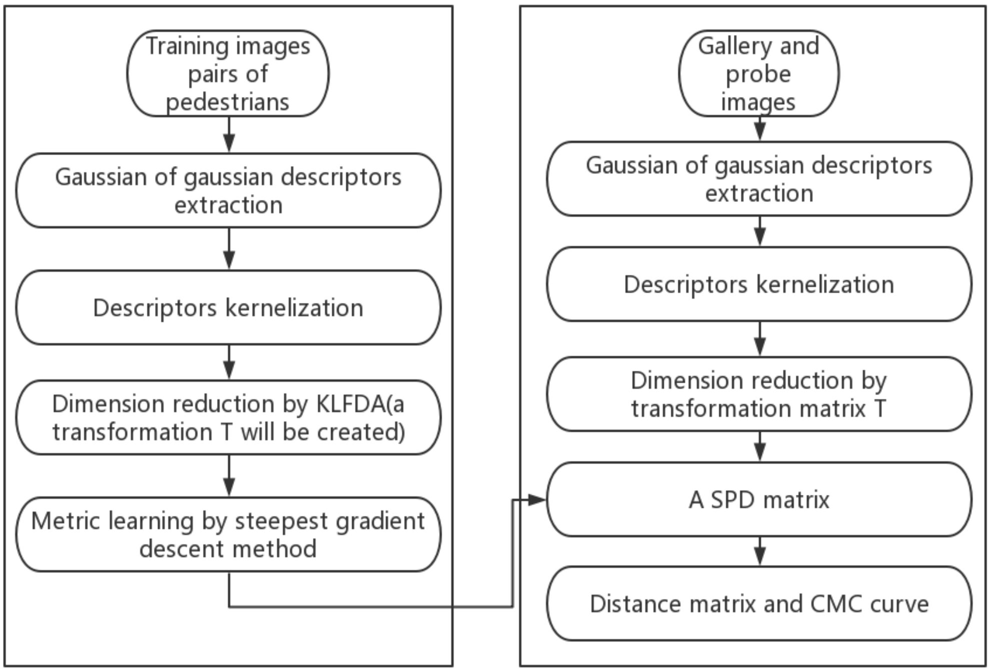

Training images pairs of pedestrians
Gaussian of gaussian descriptors extraction
Descriptors kernelization
Dimension reduction by KLFDA(a transformation T will be created)
Metric learning by steepest gradient descent method
Gallery and probe images
Gaussian of gaussian descriptors extraction
Descriptors kernelization
Dimension reduction by transformation matrix T
A SPD matrix
Distance matrix and CMC curve

FIGURE 1.6: The workflow of proposed work

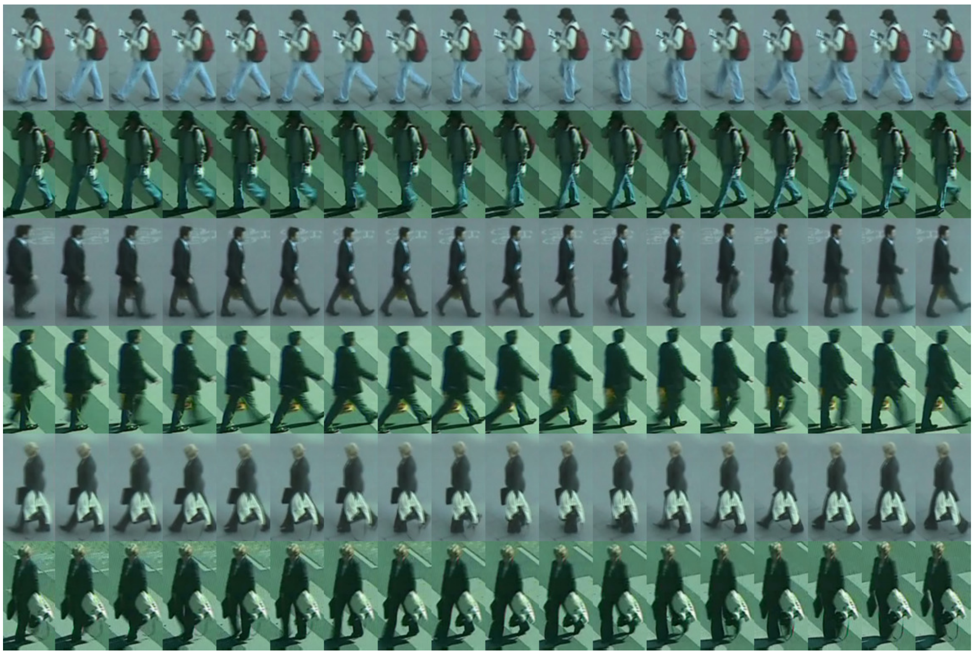


FIGURE 1.7: Samples from prid_2011 dataset

1.4 Performance measuring

There are a few measures of Re-ID, such as cumulative matching curve(CMC) curve and Receiver Operating Characteristic curve(ROC) curve. Specifically, CMC is used as a 1:m reidentification system and ROC is used for 1:1 reidentification system. In this thesis, the cumulative matching curve is used to measure Re-ID performance. The $CMC(k)$ stands for the probabilty that the right match is within the top k matches. Suppose a set of gallery $G = \{G_1, G_2, \cdots, G_m\}$ and a set of probe $P = \{P_1, P_2, \cdots, P_n\}$, for each identity P_i there should be a right match in the gallery set. However, there could be identities that appear in gallery set but not in probe set. A $m \times n$ similarity matrix can be computed. Then for each probe identity P_i, a sorted list of gallery identities can be list as $S(P_i) = \{G_1, G_2, \cdots, G_m\}$ so that their similarity with P_i descends. Suppose the right match of P_i is at the position k of $S(P_i)$, $k \leq m$, then G_i has a rank value of k. Therefore, the CMC can be calculated as

$$CMC(k) = \frac{1}{n}(\#k_l \leq k) \tag{1.1}$$

where k_l is the list of rank values of $P = \{P_1, P_2, \cdots, P_n\}$, and $\#k_l \leq k$ means the number of rank values that is smaller than k. Therefore, CMC curve always ascends and stops at 1. A perfect CMC curve is supposed to have a hight rank 1 score and approaches 1 as fast as possible.

1.5 Contribution

In this paper we have two contributions, the first is we combined the KLFDA with distance comparison learning. Instead of learning the subspace with KLFDA and computing Euclidean distance in lower dimensional space, a Mahanalobis distance based matrix is learned under the limitation that the within class distance is at least 1 unit smaller than inter class distance. Compared with those advanced metrics including cross view quadratic analysis(XQDA) [18] and Null space learning(NFST), this proposed metric learning proves to have excellent performance on VIPeR, CUHK1, prid_2011, prid_450s and GRID dataset.

Another contribution of this thesis is the influence of background subtraction on different descriptors are probed. We found that the background subtraction can improve the performance of descriptors (like HSV histogram) but can decrease the performance of certain descriptors (texture feature like LBP and HOG). This comparison is shown in Chapter 3. The reason for this is imperfect background segmentation brings about textural interference. If descriptors are color based and don't handle texture information, like HSV histogram descriptor, background segmentation can greatly improve the performance. However, if the descriptor extracts texture information, background segmentation will decrease its performance since the imperfect segmentation will cause many small black dots in foreground area, which will cause gigantic textural information variation. Because segmentation algorithm will cause different influence on various features, in this thesis, a weighted map of images is used instead of using the background segmentation.

1.6 Thesis organization

In this thesis, Chapter 2 will give a brief introduction of previous work. Chapter 3 will explain the implementation of the hierarchical gaussian descriptors used in this thesis. In Chapter 4 a detailed introduction of the kernel local fisher local discriminant analysis will be presented, and a detailed explanation will also be presented about the metric learning on the lower dimension space based on relative distance limitation learning. In Chapter 5 the used datasets and parameters and other experiment settings will be explained, and a detailed analysis of results is presented here. At last, the conclusion is given in Chapter 6.

Chapter 2

Related work

Previous work focus mainly focus on finding more discriminative descriptors and better metric learning. It's known that color and texture are the most important information in Re-ID. Most descriptors captures the local or global statistic color and texture information to characterize individuals. A brief introduction of those descriptors are given in this chapter.

2.1 Appearance descriptors

In most descriptors, the input image will be divided into a few subregions to model the complex human kinematics. Features of those subregions are extracted respectively and concatenated directly or characterized by their statistic properties. According to how those subregions are divided, there can be three kind of models, fixed-part based models, adaptive models and learned part models [33].

In fixed part models, the size of body parts are predefined. One example is in [8, 45, 31], where a silhouette is divided into a fixed number of horizontal and equal stripes, which mainly include head, torso, legs. In [16] the input image are divided into three horizontal stripes and widths of each stripe are respectively 16%, 29% and 55%. The fixed models predefine the parameters like numbers of stripes and the stripe width.

In the adaptive part models, the size of each body parts may vary to fit predefined body part models. Take [12] for an instance, the silhouette of each person is divided into three parts horizontally, which include the head, torso and legs respectively. But the width of each stripe is different for various silhouettes, and it is computed according to the symmetry and asymmetry with two operators $C(y, \sigma)$and $S(y, \sigma)$, where

$$\begin{aligned} C(y,\sigma) &= \sum d^2(p_i - \hat{p}_i) \\ S(y,\sigma) &= \sum \frac{1}{W\delta} |A(B[y, y-\delta]) - A(B[y, y+\delta])| \end{aligned} \tag{2.1}$$

Here the $C(y,\sigma)$ computes the asymmetry of two blobs and $S(y,\sigma)$ computes the difference of two areas. Then the axis between torso and legs are computed as follow

$$y_{TL} = \arg\min(1 - C(y,\sigma) + S(y,\sigma)) \tag{2.2}$$

and the axis between head and torso is computed with following equation,

$$y_{HT} = \arg\min(-S(y,\sigma)) \tag{2.3}$$

the axis divides the left and right torso is

$$j_{LR} = \arg\min(C(y,\sigma) + S(y,\sigma)) \tag{2.4}$$

This method has a relatively high performance. But one shortcoming of this model is imperfect background segmentation causes noise and errors to the axis' position.

Another part-based adaptive spatial-temporal model used in [4] characterizes person's appearance using color and facial feature. Few work exploits human face feature but in this work human face selection based on low resolution cues select useful face images to build face models. Color features capture representative color as well as the color distribution to build color model. This model handles multi-shots re-identification and it also model the color distribution variation of many consecutive frames. Besides, the facial features of this model is conditional, that is, in the absence of good face images this model is only based on color features.

Some methods based on learned part models have been proposed. Part model detectors (statistic classifiers) are trained with manually labelled human body parts images, exploiting features related with edges contained in the images. The pictorial structure is proposed in [13], and a PS model of a non-rigid body is a collection of part models with deformable configurations and connections with certain parts. The appearance of each part is separately modelled and deformable configurations are implemented with spring-like connections. This model can quantitatively describe visual appearance and model the non-rigid body. In [1] the pictorial structure body model is made up of N parts and N corresponding part detectors.

Another example of learned part model is in [5, 4], the overall human body model consists of several part models, each model is made up of a spatial model and a part filter. For each part the spatial model defines allowed arrangements of this part with respect to the bounding box. To train each model the Latent Support Vector Machine is used and four body parts are detected, namely head, left and right torso and upper legs. Compared with other models this model exploits a sequence of frames of an individual and thus captures appearance characteristics as well as the appearance variation over time.

According to the way to extract feature for each model (a whole model or part-based model), the feature can be implemented with different methods. The features can be divided into two categories, the global and local feature. The global feature refers to the feature extracted from a whole image or region, and the size of the descriptor is usually fixed. While to extract the local feature of a specified image or region, we first divide the whole image into many equal blocks and compute the feature of each block. Both descriptors may deal with color, texture and shape. The color is exploited most as the color histogram within different color space. descriptor based on texture, such as the SIFT, SURF and LBP are also widely combined to improve the performance.

Global color histogram is a frequently used global feature. For an three-channel image, like RGB image, each channel is quantized into B bins separately. The final histogram could be a multi-dimensional or mono-dimensional histogram. For instance, if $B = 8$, for multi-dimensional histogram there will be $8 \times 8 \times 8 = 512$ bins, but if we concatenate the 3 dimensional bins together the dimension can be reduced to $8 + 8 + 8 = 24$ bins while the performance of this reduced descriptor doesn?t decrease. This method can be applied on other color spaces like HSV and Lab, etc.

Local color histogram usually splits the specified model or region into many equal size blocks and compute the global feature of each block. The feature can be based on color, texture and interest points. SIFT [21] is a kind of local feature based on the interest points. The salient interest points (identifiable over rotating and scaling) are selected by the interest operator. This algorithm detects key points by computing DoG image of different scale σ with equation

$$D(x, y, \sigma) = (G(x, y, k_1\sigma) - G(x, y, k_2\sigma)) * I(x, y) \tag{2.5}$$

here $G(x, y, k_1\sigma)$ is the gaussian function with deviation $k_1\sigma$, $I(x, y)$ is the image. The DoG images are compared to find their extrema as key points. With key points localization and other processing, descriptors describing key points are created as SIFT descriptors.

Maximally stable color region (MSCR) is used in [12]. The MSCR derives from MSER (maximally stable extreme region) and detects the region with stable color cluster. It uses an agglomerative clustering algorithm to compute color clusters, and by looking at the successive time steps of the algorithm the extension of color is implemented. The detected color region is described with a nine dimensional vector containing the area, averaging color, centroid and second moment matrix. With this vector the color region detected is easy to do scale and affine transforms.

Recurrent highly-structured patches (RHSP) is also used in [12]. This feature captures patches with highly recurrent color and texture characteristics from extracted silhouette pixels. This feature is extracted with following steps, first random and probably overlapping small image patches are extracted from silhouette pixels. Then to capture those patches with informative texture the entropy of each patch (the sum of three channels' entropy) is computed, we discard those patches with entropy smaller than a specified threshold. In the next step some transforms are performed on the remaining patches to select those remain invariant to the transforms. Subsequently, the recurrence of each patch is evaluated with the LNCC(local normalized cross correlation) function. This evaluation is only performed on small region containing the patch instead of the whole image. Then the patches with high recurrence is clustered to avoid patches with similar content. Finally, the Gaussian cluster is applied to maintain the patch nearest to cluster's centroid for each cluster.

Researchers found that descriptors based on a single attribute are not robust to various datasets. That is, none of results from those descriptors outperforms other methods when tested on all datasets. A single structured descriptor can have only superior performance in a specified dataset but performs worse on other datasets. So combinations of different descriptors are exploited to improve the performance.

Moreover, 3-D model is proposed to improve Re-ID performance. A new 3-D model model called SARC3D [2] is used to represent the individual. Compared with those 2-D models, this model combines the texture and color information with their location information together to get a 3D model. This model starts with an approximate body model with single shape parameter. By precise 3-D mapping this parameter can be learned and trained with even few images (even one image is feasible). This model's construction is driven by the frontal, top and side views extracted from various videos, and for each view the silhouette of people is extracted to construct the 3-D graphical model. The final body model is sampled to get a set of vertices from previously learned graphic body model. Compared with other model, this model has a robust performance when dealing with partial occlusion, people pose and viewpoint variations since the model is based on people silhouettes from three viewpoints.

Besides, for the performance measures, for the closed set Re-ID problem, the mostly used is the cumulative matching curve(CMC). The CMC curve describes the probability of right match given a list of computed similarity score, and the first ranked ID is regarded as the matched individual. For the open-set Re-ID problem, e-ID accuracy and FAR(false accept rate) are adopted. The Re-ID accuracy is the number of probe IDs that are correctly accepted, which is expresses as true positives(TP). The FAR is expresses as the mismatches(MM) and false positives(FP). The mismatches are those probe IDs that is incorrectly matched to the galley while in fact those probe IDs exist in the gallery. The false positive is those probe IDs incorrectly matched to the gallery while they don't exist in the gallery actually. Combined descriptors are found to have better performance. Descriptors combining color and texture are most often used in re-identification. In [16] a signature called asymmetry-based histogram plus epitome(AHPE) was proposed. This work starts with a selection of images to reduce image redundancy (redundancy is caused by correlated consecutive sequences). This descriptor combines global and local statistical descriptors of human appearance, focusing on overall chromatic content via histogram and on the recurrent local patches via epitome analysis. Similar to SDALF descriptor [12], HPE descriptor consists of three components, the chromatic color histogram, the generic epitome and local epitome. The chromatic color histogram is extracted in the HSV color space, which turns to be robust to illumination changes. Here color histogram is encoded into a 36-dimensional feature space $[H = 16, S = 16, V = 4]$. Besides, the authors customize the use of epitome here by extracting generic and local epitome here.

2.2 Metric learning

Many different metric learning methods have been proposed [17, 30, 24, 42, 45, 37, 35, 38, 40, 43, 10] to get smaller intraclass distance and larger interclass distance. And Mahanalobis distance is adopted to The second step of Re-ID is to design the metric learning to match descriptors. That is, the way to compare how similiar two descriptors are. Generally, for two $d \times 1$ dimensional input vectors $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, any symmetric positive semi-definite matrix M defines a pseudo-metric with the form of $D = (\boldsymbol{x}_1 - \boldsymbol{x}_2)\boldsymbol{M}(\boldsymbol{x}_1 - \boldsymbol{x}_2)$. Many widely used distance metric exploit this rule. Previous methods includes the Euclidean distance, Bhattacharyya distance and Mahalanobis distance. The Euclidean distance, which is most common used distance, is a special case of Mahalanobis distance when the $\boldsymbol{M}$ is an identity matrix. One example of metric learning is the probabilistic relative distance comparison model proposed in [45]. This model all the possible positive person pairs with those negative pairs so that the distance of between-class distance is larger than the distance of within-class distance. Compared with other distance learning models proposed, this model solves the matrix M by an iterative optimization algorithm. Suppose z is an image of a person, the task is to identify another image z' of the same person from z'' of a different person by using a distance model $f(\cdot)$ so that $f(z, z') < f(z, z'')$. The authors convert the distance learning problem to a probability comparison problem by measuring the probability of distance between a relevant pair of images being smaller than that of a related irrelevant pair as

$$P(f(z, z') < f(z, z'')) = (1 + e^{(f(z-z')-f(z-z''))})^{-1} \tag{2.6}$$

Here the author assumes the probability of $f(z, z')$ and $f(z, z'')$ is independent, therefore, using maximal likelihood principal the optimal function can be learned as

$$f = \arg\min_{f} r(f, O) r(f, O) = -log(\Pi_{O_i} P(f(z - z') - f(z - z''))) \tag{2.7}$$

$O = \{O_i = (x_i^p - x_i^n)\}$, x_i^p, x_i^n are the pair from same person and different person respectively. The distance function $f(\cdot)$ here is parameterized as Mahalanobis distance function $f = \boldsymbol{x}^T \boldsymbol{M} \boldsymbol{x}, \boldsymbol{M} \geq 0$, here $\mathbf{M}$ is a semi-positive definite (SPD) matrix, in this way the distance function learning problem is transformed to a matrix optimization problem. The author used an iteration algorithm to compute matrix $\boldsymbol{M}$.

One shortcoming for this algorithm is it's computationally expensive because for each person it compares all the possible negative pair distance with corresponding negative pair distance.

Single shot image based person representation suffers from small sample size problem. The multi-shot Re-ID has been proposed. Since there are a sequence of images for each individual, there are much more cues to exploit. In [42], the author simplified computing of Mahananobis matrix by applying the new limitations on datasets. The author finds that when using video based person representation the difference of inter-class may be more obscure than that of still image based representation. Therefore, the author proposed the top-push distance learning. For a person video sequence, the maximal intraclass distance should be 1 unit smaller than the minimal distance of interclass distance. Another limitation of this work is the sum of all intra-class distance should be as small as possible, so the final target function is summarized as

$$\begin{aligned} f(D) = (1-\alpha) \sum_{x_i, x_j, y_i = y_j} D(x_i, x_j) + \\ \alpha \sum_{x_i, x_j, y_i = y_j} \max\{D(x_i, x_j) - \min_{y_i \neq y_k} D(x_i, x_k) + \rho, 0\} \end{aligned} \tag{2.8}$$

2.3 Other methods for Re-ID

Beside descriptors and metrics mentioned above, there are some other methods for Re-ID. Convolutional neural network have been exploited in Re-ID. One advantage of neural network Re-ID is the preprocessing of images can be skipped (We can also say the preprocessing is included in convolutional layers). The input of this structure can be straight-forward grey images or color images. To deal with multi-shots and video based re-identification neural network is proven to have better performance. Traditional neural network has too many weights to train. Convolutional neural network can avoid this problem while retaining high performance. Compared with classical neural network architecture, the convolutional neural network exploits receptive field, weights sharing and pooling technology to reduce weights number and thus decreases computational cost. In [23] the author proposes a recurrent neural network layer and temporal pooling to combine all time-steps data to generate a feature vector of the video sequence. In [7] the author proposes a multi-channel layers based neural network to jointly learn both local body parts and whole body information from input person images. In [39] a convolutional neural network learning deep feature representations from multiple domains is proposed, and this work also proposes a domain guided dropout algorithm to dropout CNN weights when learning from different datasets.

There are many other works based on convolutional neural networks. However, person re-identification may be one of the area which CNN won't work for the small sample size(SSS) problem. In most datasets, the sample size of each pedestrian is quite small. Especially in single shot Re-ID only one frame is provided in each view for each person. So Re-ID will more rely on classical machine learning.

2.4 Some state-of-the-art works

Recently many works have been proposed and improved Re-ID performance by much margin. In this section those advanced descriptors and metrics are introduced. **Cross view quadratic discriminant analysis**(XQDA) is proposed in [18]. Suppose the sample difference $\Delta = \boldsymbol{x}_i - \boldsymbol{x}_j$, where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are two feature vectors. Δ is called intrapersonal difference when their label satisfy $y_i = y_j$ and extrapersonal difference when $y_i \neq y_j$. Respectively two the intrapersonal and interpersonal variation can be defined as Ω_I and Ω_E, the authors convert Re-ID problem to distinguish Ω_I and Ω_E. In [25] each one of intrapersonal and interpersonal class is modelled with a multivariate gaussian distribution, and in [25] it has been proved that both Ω_I and Ω_E have zero mean. Under the zero-mean distribution, the probability of observing Δ in Ω_I and the probability of observing Δ in Ω_E can be denoted as

$$\begin{aligned} P(\Delta|\Omega_I) &= \frac{1}{(2\pi)^{d/2}|\Sigma_I|^{1/2}} \exp^{\frac{-1}{2}\Delta^T\Sigma_I^{-1}\Delta} \\ P(\Delta|\Omega_E) &= \frac{1}{(2\pi)^{d/2}|\Sigma_E|^{1/2}} \exp^{\frac{-1}{2}\Delta^T\Sigma_E^{-1}\Delta} \end{aligned} \tag{2.9}$$

where Σ_I and Σ_E are the covariance matrix of Ω_I and Ω_E, then the probability ratio between the interpersonal pairs and intrapersonal pairs can be denoted as

$$\begin{aligned} r(\Delta) = \frac{P(\Delta|\Omega_E)}{P(\Delta|\Omega_I)} &= \frac{\frac{1}{(2\pi)^{d/2}|\Sigma_E|^{1/2}} \exp^{\frac{-1}{2}\Delta^T\Sigma_E^{-1}\Delta}}{\frac{1}{(2\pi)^{d/2}|\Sigma_I|^{1/2}} \exp^{\frac{-1}{2}\Delta^T\Sigma_I^{-1}\Delta}} \\ r(\Delta) &= C \exp^{\frac{-1}{2}\Delta^T(\Sigma_E^{-1}-\Sigma_I^{-1})\Delta} \end{aligned} \tag{2.10}$$

C is the constant term, by taking log and deserting the constant term, we have

$$r(\Delta) = \Delta^T(\Sigma_I^{-1} - \Sigma_E^{-1})\Delta = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T(\Sigma_I^{-1} - \Sigma_E^{-1})(\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{2.11}$$

In [18] a subspace W is learned so that

$$r(\Delta) = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T W({\Sigma_I}'^{-1} - {\Sigma_E}'^{-1})W^T(\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{2.12}$$

and ${\Sigma_I}' = W^T\Sigma_I W, {\Sigma_E}' = W^T\Sigma_E W$. Therefore, a subspace $M(W) = W({\Sigma_I}'^{-1} - {\Sigma_E}'^{-1})W^T$ is learned in this work.

Null Foley-Sammon transform In [43] a null space is proposed so that with this space the intraclass points collapse to a same point in the null space while interclass points are projected to different points. Given the within class scatter $\boldsymbol{S}^w$ and between class scatter $\boldsymbol{S}^b$, an optimal projection matrix $\boldsymbol{W}$ is computed so that

$$\begin{aligned} \boldsymbol{w}_i^T\boldsymbol{S}^w\boldsymbol{w}_i &= 0 \\ \boldsymbol{w}_i^T\boldsymbol{S}^b\boldsymbol{w}_i &> 0 \end{aligned} \tag{2.13}$$

$\boldsymbol{w}_i$ is the i_{th} column in $\boldsymbol{W}$.

It can be noticed that like many other metric learnings, XQDA and NFST are transformed into matrix decomposition and eigenvalue selection problem. In this paper, those two metrics are used to compare with proposed metric. In [22] it has been shown GOG + XQDA outperforms many other combinations including MetricEnsemble [29], SCNCD [41], SemanticMethod [34], etc. In [43] it has been shown LOMO + NFST outperforms metrics including LMNN [37], KCCA [38], ITML [10], KLFDA[35], MFA [40], KISSME [17], SimilarityLearning [6], SCNCD [41], Mid-level Filters [44] and Improved Deep [11]. Based on the result that XQDA and NFST outperform other metrics, in this thesis, XQDA and NFST are used to compare with proposed metric learning.

Chapter 3

Descriptors extraction and dimension reduction

In person re-identification, it's very important to choose robust descriptor to represent person. A good descriptor should be robust to variations of illumination, viewpoint, and camera color response. Most descriptors tries to seize the color and texture information. In this chapter, we will first introduce some basic descriptors and compare their performance on VIPeR dataset, then a detailed introduction of hierarchical descriptor will be presented in the coming section.

3.1 Basic color and textural features

3.1.1 Color histogram descriptors on different color space

Histogram descriptor extracts color statistics information of input images. A popular histogram extracting method is to divide input image into a few horizontal stripes and extract color histogram of each stripe, then they are concatenated to consist of histogram descriptor of the whole image. Color space selection has much influence on descriptor performance. HSV color space is very common in computer vision and image processing area for target detection and tracking. The HSV descriptor has better performance than RGB histogram descriptor since HSV color separates image intensity from color information. Thus HSV color space is more robust to illumination variation. An unsupervised CMC performance comparison among different color spaces on VIPeR dataset is given in figure 3.1.1. In this comparison camera a views are used as probe set and camera B views are used for gallery set. We can find

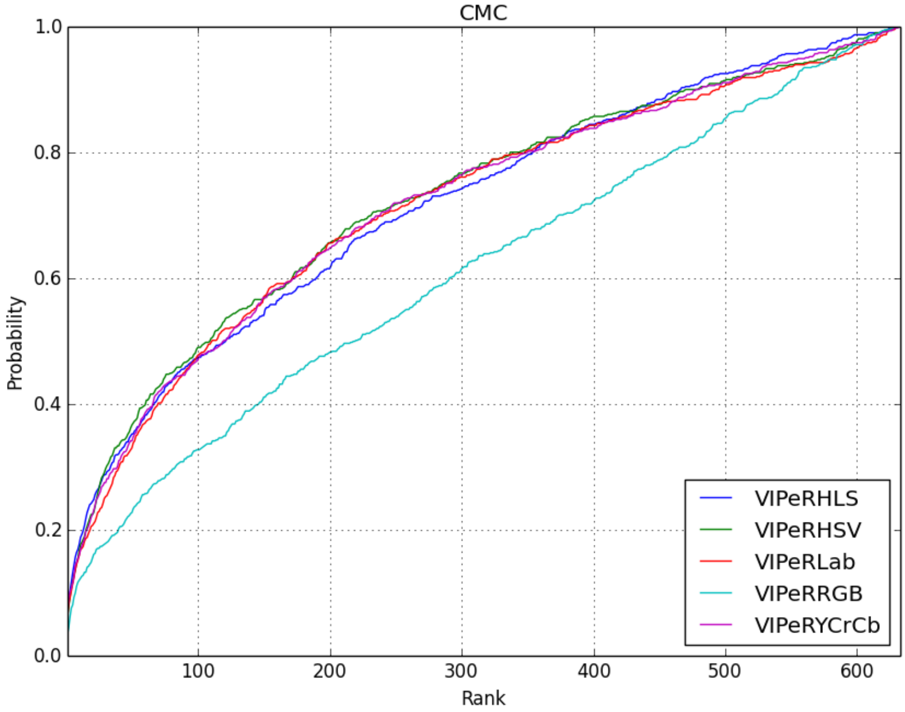


FIGURE 3.2: A CMC comparison of color histogram on different color spaces

that those color spaces separating intensity information outperform RGB color space and HSV outperforms all other color spaces.

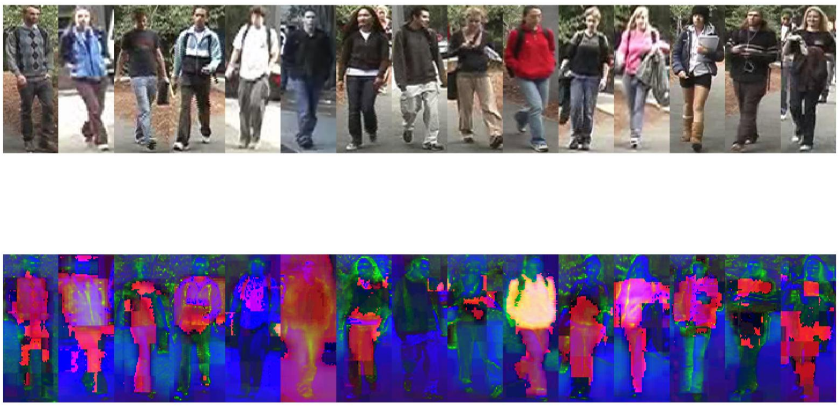


FIGURE 3.1: RGB and HSV visual comparison, the first row is RGB and second row is HSV for same views

Analysis of histogram based descriptor The performance of histogram descriptors suffers from ignoring the spatial information. Since it doesn't consider the relative distribution of color, images with same kind color patches but different distribution may have the same histogram descriptor. One example is shown in figure 3.1.1.

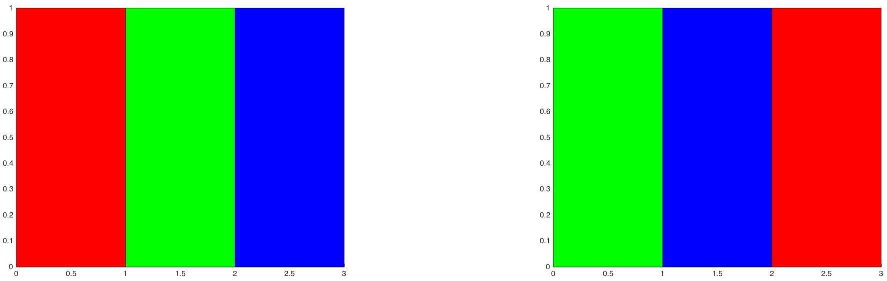


FIGURE 3.3: A comparison of two patches with same entropy but different color distribution

3.1.2 Local binary pattern(LBP)

Local binary pattern [27, 26] extracts the texture information with efficient computing and has been used on people detection and recognitions. Figure 3.1.2 is an example of LBP example. by thresholding neighbour pixel of center pixel, the pixels are transformed into a binary integer. There are many extended LBP like tLBP[], VLBP[], OCLBP[], and LBP is well known for its robustness to monotonic illumination variation.

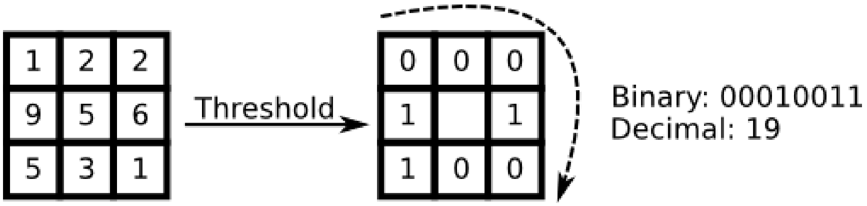


FIGURE 3.4: An LBP example, by thresholding the neighbour pixels the pixels are transformed into a binary number

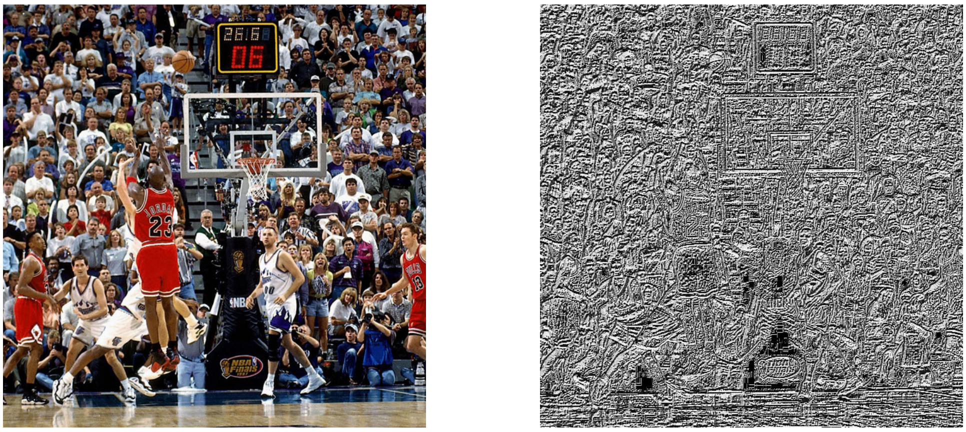


FIGURE 3.5: One LBP example

3.1.3 Histogram of oriented gradients(HOG)

The HOG [9] descriptor also extracts textural information of images by gradient computing. A brief introduction about its gradient computation is presented here, more details can be referenced in [9]. In HOG feature it computes the gradient of input intensity image $I(x, y)$ by equations

$$\begin{aligned} I_x &= \frac{\partial I}{\partial x}, \\ I_y &= \frac{\partial I}{\partial y}, \end{aligned} \tag{3.1}$$

the gradient can be computed fast by some discrete derivative masks below, like 1-D Sobel masks:

$$\begin{aligned} Centered : M_c &= [-1, 0, 1] \\ Uncentered : M_{uc} &= [-1, 1] \end{aligned} \tag{3.2}$$

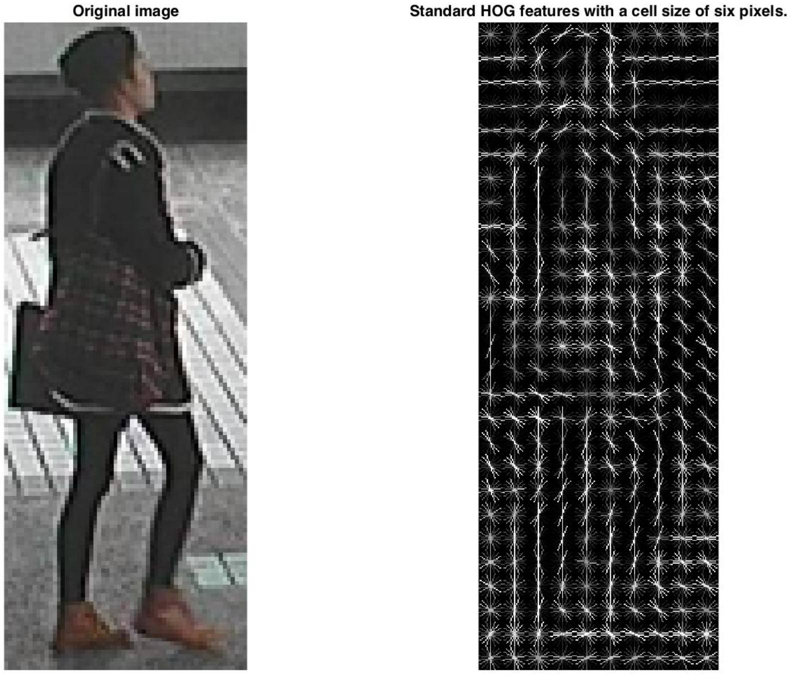
or 2-D Sobel masks:

$$\begin{aligned} D_x &= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \\ D_y &= \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{3.3}$$

or 3×3 Sobel masks:

$$
\begin{aligned}
S_x &= \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \\
S_y &= \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}
\end{aligned}
\tag{3.4}
$$

Using different masks will result different performance. Besides, gaussian smoothing is often performed before gradient computing. It has been shown that using 1-D Sobel without gaussian smoothing has the best performance. A HOG feature demo is shown below.



Original image

Standard HOG features with a cell size of six pixels.

FIGURE 3.6: A demo of HOG feature with a cell size of six pixels

3.2 Influence of background segmentation on different basic descriptors

Many works try to minimize impact of background noise of pedestrians' image. It's easier to automatically segment foreground from sequential videos than a single frame. In [12] the author provides foreground mask for all images following the algorithm in [15]. Some of those segmented foreground are shown in figure [] and it's obvious that certain bogy parts like head and feet are lost. To compare those loss's impact on color and textural descriptors, a comparison of foreground segmentation on HSV color histogram descriptors and local binary pattern(LBP) is given in figure 3.2, figure 3.2 and figure 3.2.

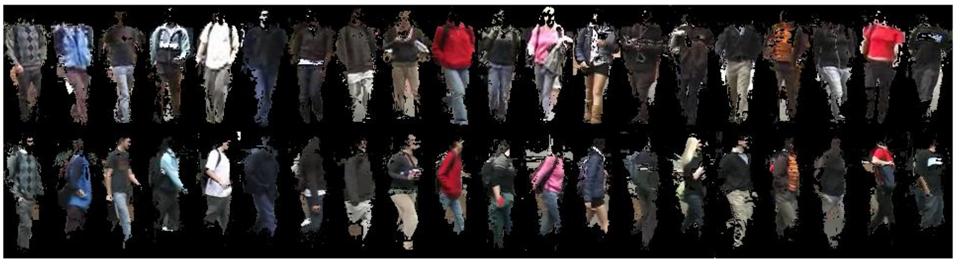


FIGURE 3.7: Foreground segmentation of individuals from VIPeR

We can find that foreground segmentation decreases LBP and HOG's performance but increases HSV color histogram's performance on VIPeR dataset greatly. The reason for this is imperfect foreground segmentation causes body parts(like head and feet) loss and small black patches in torso and legs, and for some individuals a part of background scene is regarded as foreground. Since HSV color histogram doesn't handle spatial distribution but only color entropy, foreground segmentation improves its performance greatly. But since LBP and HOG handle texture for each sample patch, its performance suffers from those body parts loss and little black patches. What's more, we can infer that imperfect foreground segmentation will also decrease other textural feature's performance.

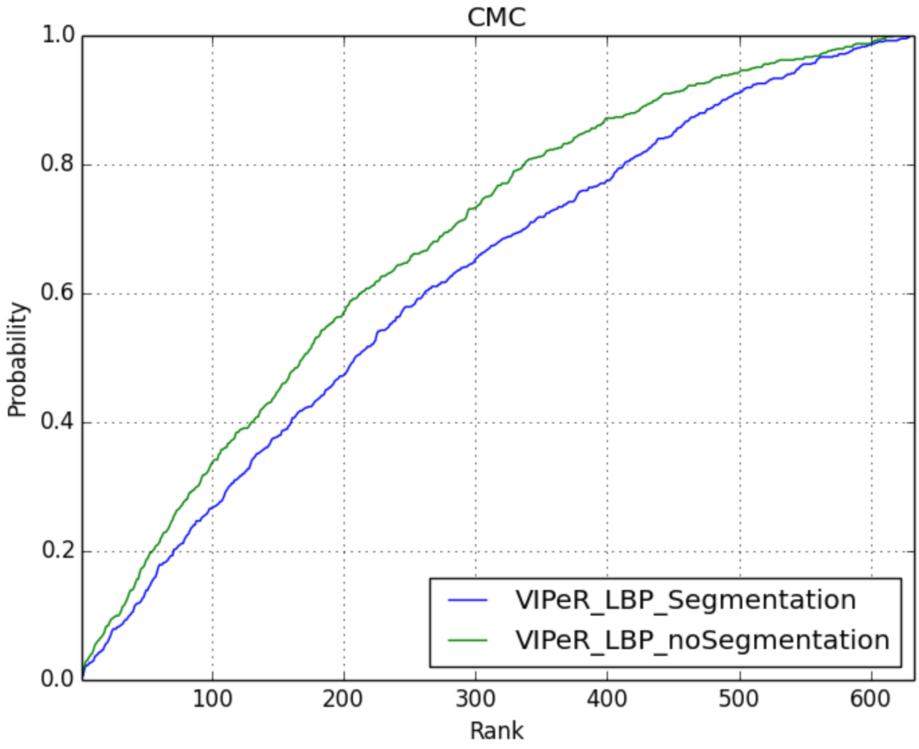


FIGURE 3.8: A CMC comparison of foreground segmentation on LBP feature tested on VIPeR

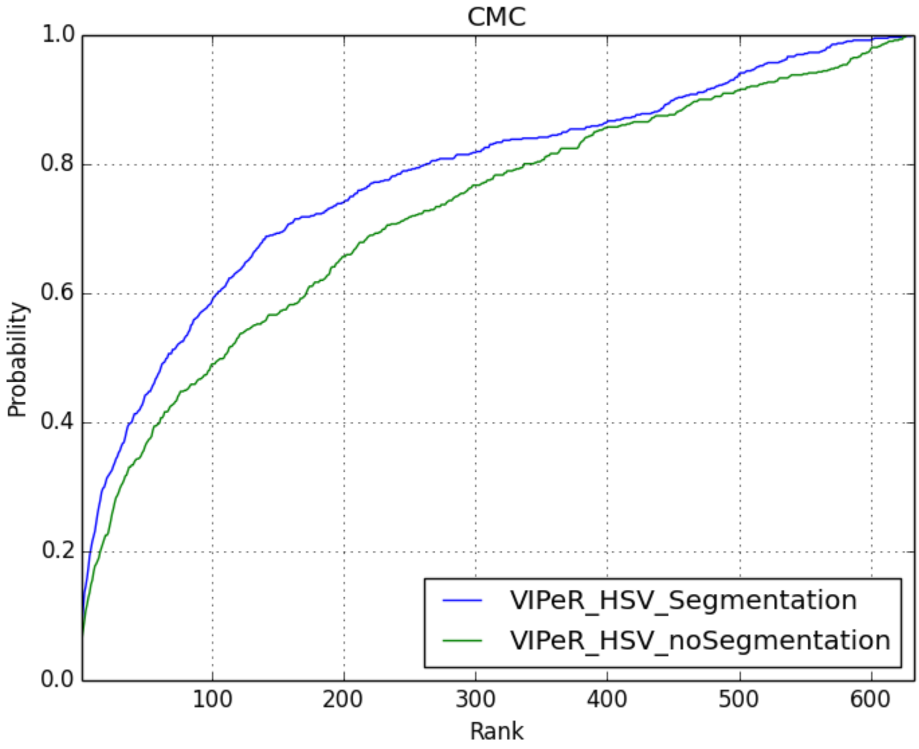


FIGURE 3.9: A CMC comparison of foreground segmentation on HSV histogram descriptor tested on VIPeR

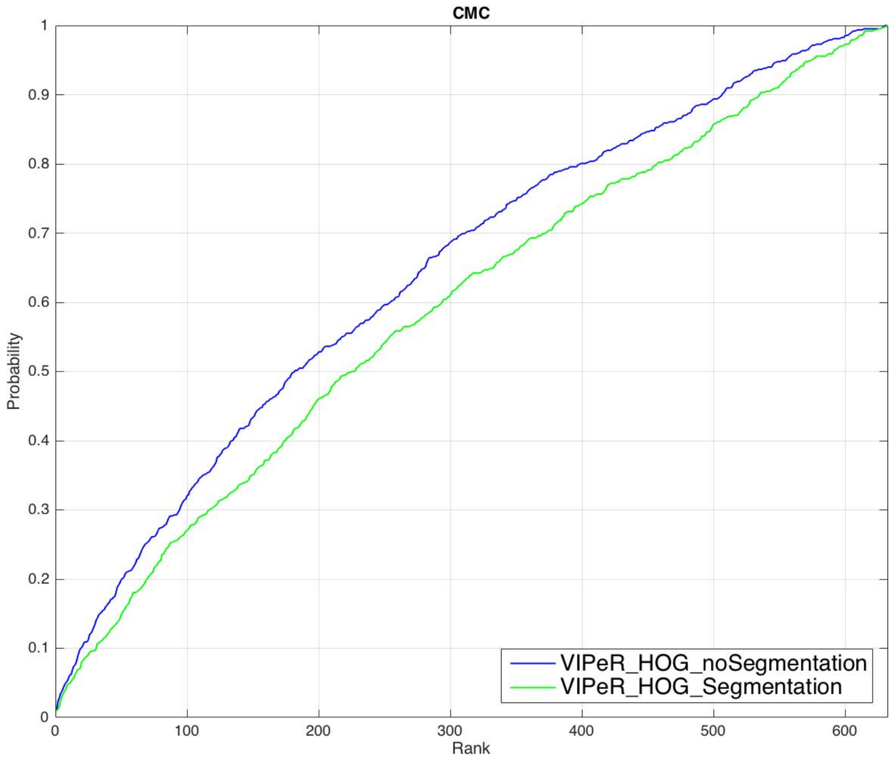


FIGURE 3.10: A CMC comparison of foreground segmentation on HOG feature tested on VIPeR

3.3 The hierarchical gaussian descriptor

The hierarchical gaussian descriptor is proposed by in [22], this descriptor uses a two-level gaussian distribution to model an individual. This descriptor densely sample the image and model each hierarchical structure with gaussian distribution and has outperformed many other works. Firstly it divides the image into a few overlapping horizontal slides, and in each slide, dense sampling patches are made with certain size. So there is a two-level structure in this image, small patches and slides. Then by model each level with gaussian model we can get a robust representation of the individual.

3.3.1 Handling the background

In last section the impact of background subtraction on different features' performance have been probed. Based on the result that imperfect foreground segmentation will decrease textural feature's performance, and in hierarchical gaussian descriptor there is no histogram based feature computing. When modelling the region gaussian a weighted map is computed for each patch with equation

$$N(x;\mu_0,\sigma_0) = \frac{1}{\sigma_0\sqrt{2\pi}} \exp^{\frac{(x-\mu_0)^2}{\sigma_0^2}} \tag{3.5}$$

and here $\mu_0 = \frac{W_0}{2}, \sigma_0 = \frac{W_0}{4}$, W_0 is the number of patches in horizontal direction.

3.3.2 Single pixel modelling

In this hierarchical model, it is very important to have a full representation for every single pixel. To fully characterize single pixel, a d dimensional vector is used to represent it. In this vector, there could be any predefined properties like coordinates, color values, texture and filter response. Suppose the original image is in RGB color space, the gaussian of gaussian descriptor uses a 8-dimensional vector $\boldsymbol{f}_i$, and $\boldsymbol{f}_i = (y, M_0, M_{90}, M_{180}, M_{270}, R, G, B)$. The y component is the y coordinate of pixel, and $M_{\{\theta \in 0^o, 90^o, 180^o, 270^o\}}$ is the quantized gradient information in 4 directions. The last three component is the color value is specified color space.

In all the benchmark dataset, all the images are cropped with a bounding box well suited the individual, and the pedestrian in an image can be at left or right of center, while in the vertical direction the head and feet of pedestrian is very close the image edge. For each pixel, the y coordinate is more correlated than x coordinate, so only y coordinate is chosen for pixel modelling.

Then the M is to characterize the texture with the gradient histogram. Different M values is the magnitude of gradient in every direction. Firstly the gradient in x and y direction are computed by two gradient filters $\mathrm{h_x}$ and $\mathrm{h_y}$, and we have

$$\begin{aligned} h_x &= [-1, 0, 1] \\ h_y &= -h_x' \end{aligned} \tag{3.6}$$

Then by convolve those two filters with the intensity image I, the horizontal and vertical gradient I_x, I_y can be computed, so the orientation and magnitude can be computed by following equations:

$$\begin{aligned} O(i,j) &= (\arctan(I_y(i,j)/I_x(i,j)) + \pi) * 180/\pi \\ M(i,j) &= \sqrt{(I_x(i,j)^2 + I_y(i,j))^2} \end{aligned} \tag{3.7}$$

The orientation are quantized into four bins by a soft voting algorithm [28]. For each pixel its corresponding gradient orientation is decided by its nearest bin's direction. To make the descriptor focus on the gradient components with high values, the gradient and orientation are multiplied as follow,

$$M_\theta = M O_\theta, \tag{3.8}$$

To model the patch with a multi-variate gaussian distribution, we have to estimate its mean value and the covariance matrix. A multi-variate gaussian model has the form

$$G(\boldsymbol{f}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{\exp^{(\frac{1}{2}(\boldsymbol{f}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{f}_i - \boldsymbol{\mu}))}}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|} \tag{3.9}$$

where $\boldsymbol{\mu}$ is the estimated mean value, and $\boldsymbol{\Sigma}$ is the estimated covariance matrix.

To estimate the parameters for this gaussian model based on sampled patches pixel features, the maximal likelihood estimate(MLE) is used. According MLE algorithm, we have the following estimated parameters

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{f}_i, \tag{3.10}$$

$$\boldsymbol{\Sigma} = \frac{1}{n-1}\sum_{i=1}^{n} (\boldsymbol{f}_i - \boldsymbol{\mu})(\boldsymbol{f}_i - \boldsymbol{\mu})^T, \tag{3.11}$$

where n is the number of pixels in current patch. When the gaussian model is computed, the next step is to model all the patch gaussians. But it's a complex problem to directly model those multivariate gaussian functions. So some transformation will be operated on estimated parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$.

3.3.3 Integral image for fast region covariance computation

To compute estimated parameters for all those overlapping small patches, the time complexity of computing one by one patch is gigantic because there are many repeating computations. To compute the estimated covariance matrix $\boldsymbol{\Sigma}$ for every small patch with size of $W \times H$, the integral image is used to reduce time complexity. The integral image [36] is a intermediate representation to fast compute rectangle area sum in an image. Each pixel value in integral image is the sum of all the pixels inside the rectangle bounded by current pixel and the upper left pixel. That is, the integral image S(x, y) for image I(x, y) is

$$S(x', y') = \sum_{x<x', y<y'} I(x, y), \tag{3.12}$$

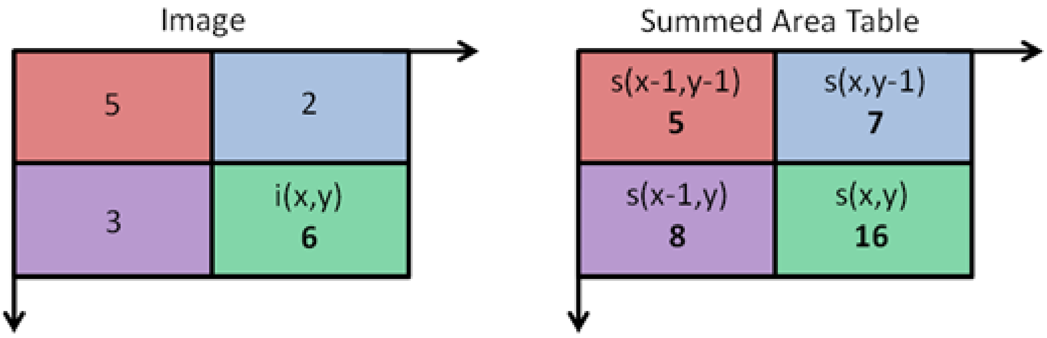


FIGURE 3.11: Integral image

By using integral image any rectangular region sum can be computed in constant time.

To compute the covariance matrix of a certain rectangle area in a $W \times H \times d$ dimensional feature tensor F, suppose $\boldsymbol{I}_F$ is the $W \times H \times d$ tensor of integral images of F, we have

$$\boldsymbol{I}_F(x', y', i) = \sum_{x<x', y<y'} F(x, y, i), i = 1 \ldots d \tag{3.13}$$

and suppose the $\boldsymbol{C}(x', y', i, j)$ is the $W \times H \times d \times d$ tensor of second order integral images, we have

$$\boldsymbol{C}(x', y', i, j) = \sum_{x<x', y<y'} F(x, y, i) F(x, y, j), i, j = 1 \ldots d. \tag{3.14}$$

let $\boldsymbol{I}_{x,y}$ be the d dimensional vector in $\boldsymbol{I}_F$, $\boldsymbol{C}(x, y)$ be the $d \times d$ dimensional matrix in $\boldsymbol{C}$,

$$\begin{aligned} \boldsymbol{I}_{x,y} &= [\boldsymbol{I}_F(x', y', 1) \dots \boldsymbol{I}_F(x', y', d)]^T \\ \boldsymbol{C}_{x,y} &= \begin{bmatrix} \boldsymbol{C}(x, y, i, 1) & \cdots & \boldsymbol{C}(x, y, 1, d) \\ & \ddots & \\ \boldsymbol{C}(x, y, d, 1) & \cdots & \boldsymbol{C}(x, y, d, d) \end{bmatrix} \end{aligned} \tag{3.15}$$

Then for any rectangule regions $R(x', y'; x'', y'')$, where (x', y') is the upper left coordinate and (x", y") is the lower right coordinate, the covariance matrix can be compute as

$$\begin{aligned} \boldsymbol{C}_R(x', y'; x'', y'') = \frac{1}{n-1}[\boldsymbol{C}_{x'',y''} + \boldsymbol{C}_{x',y'} - \boldsymbol{C}_{x'',y'} - \boldsymbol{C}_{x',y''} \\ -\frac{1}{n}(\boldsymbol{I}_{x'',y''} + \boldsymbol{I}_{x'',y''} - \boldsymbol{I}_{x',y''} - \boldsymbol{I}_{x'',y'})(\boldsymbol{I}_{x'',y''} + \boldsymbol{I}_{x'',y''} - \boldsymbol{I}_{x',y''} - \boldsymbol{I}_{x'',y'})^T] \end{aligned} \tag{3.16}$$

where n is the number of feature vector in F, and $n = (x'' - x')(y'' - y')$. By creating the integral image the covariance of any rectangular area in F can be computed in $O(d^2)$ time.

When all patches in a region are computed, the same process is repeated to compute the region gaussian.

3.3.4 Riemannian manifold based SPD transformation

As described before this hierarchical gaussian descriptor is a stochastic feature, so operations like computing mean and covariance need to be operated on previous summarized gaussian distributions. Mean and covariance operation in Euclidean space can not be directly finished on previous estimated gaussian functions. A transformation is needed to make stochastic summarization feasible on patch gaussian function. In fact, the multivariate gaussian model is a Riemannian manifold and can be embedded into a semi positive definite matrix(SPD) space. The gaussian function is mapped into a vector space with two steps mapping. A d dimensional multivariate gaussian function can be mapped into a $d + 1$ dimensional SPD_+ space. According to [20], the mapping can be denoted as

$$G(\boldsymbol{x}_i; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \sim \boldsymbol{P}_i = |\boldsymbol{\Sigma}_i|^{1/(d+1)} \begin{bmatrix} \boldsymbol{\Sigma}_i + \boldsymbol{\mu}_i \boldsymbol{\mu}^T & \boldsymbol{\mu}_i \\ \boldsymbol{\mu}_i^T & 1 \end{bmatrix} \tag{3.17}$$

The covariance matrix $\mathbf{\Sigma}_i$ can be singular for small number of pixels within the patch, to avoid this problem a regular factor λ is added to $\mathbf{\Sigma}_i$ so that $\mathbf{\Sigma}_i = \mathbf{\Sigma}_i + \lambda \boldsymbol{I}$.

After this mapping, the $n+1$ dimensional SPD matrix needs to be transformed into a vector. The matrix logarithm is used to transform it to tangent space. A $d+1$ dimensional SPD matrix can be mapped as a $d*(d+3)/2+1$ vector, which can be denoted as $SPD_i^+ \sim \boldsymbol{p}_i = vec(log(\boldsymbol{P}_i))$. Since $\boldsymbol{P}_i$ is a positive symmetric matrix, it can be compressed by half that only the upper triangular elements are preserved. To ensure its norm-1 remain the same after compression, the magnitude of off-diagonal elements in $\boldsymbol{P}_i$ are timed by $\sqrt{2}$. Let $\boldsymbol{Q} = \log \boldsymbol{P}_i$, we have

$$\boldsymbol{p}_i = [\boldsymbol{Q}_{1,1}, \sqrt{2}\boldsymbol{Q}_{1,2}, \sqrt{2}\boldsymbol{Q}_{1,3}, \cdots, \sqrt{2}\boldsymbol{Q}_{1,d+1}, \tag{3.18}$$

$$\boldsymbol{Q}_{2,2}, \sqrt{2}\boldsymbol{Q}_{2,3,}, \cdots, \sqrt{2}\boldsymbol{Q}_{2,d+1,}, \cdots, \boldsymbol{Q}_{d+1,d+1,}] \tag{3.19}$$

In this thesis,

With the Gaussian parameters extracted in each region, the same transformation is operated on them. Then all horizontal slides' descriptor are concatenated to get the whole descriptor for the whole image.

Dimension analysis It has been known that combination of descriptors of different color space can greatly improve re-ID performance. In this project, the hierarchical gaussian descriptor in RGB color space is the base descriptor. Descriptors in three more color space {HSV, Lab, nRGB} are extracted. The nRGB color space is calculated as

$$\begin{aligned} nR &= \frac{R}{R+G+B}, \\ nG &= \frac{G}{R+G+B}, \\ nR &= \frac{B}{R+G+B}, \end{aligned} \tag{3.20}$$

since nB can be calculated with nR and nG, in this color space only the first two channel values are used to reduce redundancy. Therefore, for color space {RGB, HSV, Lab, nRGB}, the corresponding dimension of pixel feature is {8, 8, 8, 7}. After the matrix to vector transformation, the dimension of patch gaussian vector of each channel is {45, 45, 45, 36}. Again after the patch gaussian to region gaussian transformation, the dimension of each channel is {1081, 1081, 1081, 703}. Suppose there are 7 horizontal slides in each image, the dimension of concatenated descriptor of each channel is {7567, 7567, 7567, 4921}. If four color space are all used, the dimension is the sum of each channel as 27622.

Chapter 4

Dimension reduction and Mahanalobis distance learning

To deal with high dimensional descriptors, dimension reduction are firstly performed by kernel local fisher discriminant analysis (KLFDA). Then Mahanalobis distance metric learning based on limitations between interclass and intraclass distance are operated on dimension reduced data.

4.1 Kernel local fisher discriminant analysis

The extracted hierarchical gaussian descriptors have high dimension, it's intractable to learn a SPD matrix with such a high dimension. Dimension reduction is required to learn a subspace. Among those methods to reduce dimension, principal component analysis (PCA) is often used. However, PCA is an unsupervised dimension reduction and may have a low performance for those reasons, (1), PCA is to maximize the variance of dimension reduced data, and as a unsupervised method it doesn't has a full consideration of the the relation of between and within classes, it is very likely that the descriptors of different classes can be mixed up after the dimension reduction; (2) PCA may suffer from the small sample size problem. In some Re-ID datasets, there may be two or less images for each pedestrian in each viewpoint (like VIPeR), if the dimension of descriptor is much bigger than sample size, much information can be lost with PCA. In this thesis, the kernel local fisher discriminant analysis (KLFDA) is used to reduce dimension.

KLFDA is the kernel version of LFDA, and LFDA is a combination of Fisher discriminant analysis [30] and and the locality preserving projection [14] and kernel method. A brief introduction of FDA, LPP and kernel method is introduced below.

4.1.1 Fisher discriminant analysis (FDA)

FDA is a supervised dimension reduction and its input contains the class labels. For a set of d-dimensional observations $\boldsymbol{x}_i$, where $i \in \{1, 2, \cdots, n\}$, the label $l_i \in \{1, 2, \cdots, l\}$. Two matrix are defined as the intraclass scatter matrix $\boldsymbol{S}^{(w)}$ and between class scatter matrix $\boldsymbol{S}^{(b)}$,

$$\begin{aligned} \boldsymbol{S}^{(w)} &= \sum_{i=1}^{l} \sum_{j:l_j=i} (\boldsymbol{x}_j - \boldsymbol{\mu}_i)(\boldsymbol{x}_j - \boldsymbol{\mu}_i)^T \\ \boldsymbol{S}^{(b)} &= \sum_{i=1}^{l} n_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \end{aligned} \tag{4.1}$$

where the $\boldsymbol{\mu}_i$ is the mean of samples whose label is i, and $\boldsymbol{\mu}$ is the mean of all samples,

$$\begin{aligned} \boldsymbol{\mu}_i &= \frac{1}{n_i} \sum \boldsymbol{x}_i, \\ \boldsymbol{\mu} &= \frac{1}{n} \sum \boldsymbol{x}_i \end{aligned} \tag{4.2}$$

The Fisher Discriminant Analysis transform matrix $\boldsymbol{T}$ can be represented as

$$\boldsymbol{T} = \arg\max \frac{\boldsymbol{T}^T \boldsymbol{S}^{(b)} \boldsymbol{T}}{\boldsymbol{T}^T \boldsymbol{S}^{(w)} \boldsymbol{T}} \tag{4.3}$$

This equation can be solved by Lagrange multiplier method, we define a Lagrange function

$$L(\boldsymbol{t}) = \boldsymbol{t}^T \boldsymbol{S}^{(b)} \boldsymbol{t} - \lambda(\boldsymbol{t}^T \boldsymbol{S}^{(w)} \boldsymbol{t} - 1) \tag{4.4}$$

Then the differential respect to $\boldsymbol{t}$ is

$$\frac{\partial L(\boldsymbol{t})}{\partial \boldsymbol{t}} = 2\boldsymbol{S}^{(b)} \boldsymbol{t} - 2\lambda \boldsymbol{S}^{(w)} \boldsymbol{t} \tag{4.5}$$

let

$$\frac{\partial L(\boldsymbol{t})}{\partial \boldsymbol{t}} = 0 \tag{4.6}$$

we can get

$$\boldsymbol{S}^{(b)} \boldsymbol{t}_i = \lambda \boldsymbol{S}^{(w)} \boldsymbol{t}_i \tag{4.7}$$

here $\boldsymbol{t}_i$ is the i_{th} column of $\boldsymbol{T}$, and the optimization problem is converted to a eigenvalue decomposition problem.

Fisher discriminant analysis tries to minimize the intraclass scatter matrix while maximize the interclass scatter matrix, and $\boldsymbol{T}$ is computed by the eigenvalue decomposition. $\boldsymbol{T}$ can be represented as the set of all the corresponding eigenvectors, as $\boldsymbol{T} = (\boldsymbol{t}_1, \boldsymbol{t}_2, \cdots, \boldsymbol{t}_k)$.

FDA has a form similar with signal and noise ratio, however, the FDA dimension reduction may have poor performance for it doesn't consider the locality of data. An example of this is the multimodality [30]. Multimodality is the case many clusters are formed in the same class.

4.1.2 Locality preserving projection (LPP)

In [14] locality preserving projection (LPP) is proposed to exploit data locality. An affinity matrix is created to record the affinity of sample $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, typically the range of elements in $\boldsymbol{A}_{i,j}$ is $[0, 1]$. There are many manners to define a $n \times n$ affinity matrix $\boldsymbol{A}$, usually two sample points with a smaller distance has a higher affinity value than those with bigger distance value. One of them is if $\boldsymbol{x}_i$ is within k-nearest neighbours of $\boldsymbol{x}_j$ then $\boldsymbol{A}_{i,j} = 1$ otherwise $\boldsymbol{A}_{i,j} = 0$.

Another diagonal matrix $\boldsymbol{D}$ can be defined that each diagonal element is the sum of corresponding column in $\boldsymbol{A}$,

$$\boldsymbol{D}_{i,i} = \sum_{j=1}^{n} \boldsymbol{A}_{i,j} \tag{4.8}$$

then the LPP transform matrix is defined as follow,

$$\boldsymbol{T}_{LPP} = \underset{\boldsymbol{T} \in \boldsymbol{R}^{d \times m}}{\arg\min} \frac{1}{2} \sum_{i,j=1}^{n} \boldsymbol{A}_{i,j} ||\boldsymbol{T}^T \boldsymbol{x}_i - \boldsymbol{T}^T \boldsymbol{x}_j|| \tag{4.9}$$

so that $\boldsymbol{T}^T \boldsymbol{X} \boldsymbol{D} \boldsymbol{X}^T \boldsymbol{T} = \boldsymbol{I}$. Suppose the subspace has a dimension of m, then LPP transform matrix T can be represented as

$$\boldsymbol{T}_{LPP} = \{\boldsymbol{\phi}_{d-m+1} | \boldsymbol{\phi}_{d-m+2} | \cdots \boldsymbol{\phi}_d\}$$

And each $\boldsymbol{\phi}$ in T is the eigenvector of following fomula,

$$\boldsymbol{X} \boldsymbol{L} \boldsymbol{X}^T \boldsymbol{\phi} = \gamma \boldsymbol{X} \boldsymbol{D} \boldsymbol{X}^T \tag{4.10}$$

where γ is corresponding eigenvalue of $\boldsymbol{\phi}$, and $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$.

4.1.3 Local fisher discriminant analysis

LFDA [30] combines FDA and LPP and have better performance. The key in LFDA is it assigns weights to elements in $\boldsymbol{A}^{(w)}$ and $\boldsymbol{A}^{(b)}$, so that,

$$\begin{aligned} \boldsymbol{S}^{(w)} &= \frac{1}{2}\sum_{i=1}^{l}\sum_{j:l_j=i} \boldsymbol{A}^{w}_{i,j}(\boldsymbol{x}_j - \boldsymbol{\mu}_i)(\boldsymbol{x}_j - \boldsymbol{\mu}_i)^T \\ \boldsymbol{S}^{(b)} &= \frac{1}{2}\sum_{i=1}^{l} \boldsymbol{A}^{b}_{i,j}(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \end{aligned} \tag{4.11}$$

where

$$\begin{aligned} \boldsymbol{A}^{(w)}_{i,j} &= \begin{cases} \boldsymbol{A}_{i,j}/n_c & y_i = y_j \\ 0 & y_i \neq y_j \end{cases} \\ \boldsymbol{A}^{(b)}_{i,j} &= \begin{cases} (\frac{1}{n} - \frac{1}{n_c})\boldsymbol{A}_{i,j} & y_i = y_j \\ \frac{1}{n} & y_i \neq y_j \end{cases} \end{aligned} \tag{4.12}$$

where y_i is the class label of sample point $\boldsymbol{x}_i$. So the transformation matrix T_LFDA can be computed by equation

$$\boldsymbol{T}_{LFDA} = \arg\min_{\boldsymbol{T}}(\frac{\boldsymbol{T}^T\boldsymbol{S}^{(b)}\boldsymbol{T}}{\boldsymbol{T}^T\boldsymbol{S}^{(w)}\boldsymbol{T}}) \tag{4.13}$$

Again this problem can be solved by eigenvalue decomposition by equation 4.1.1.

When applying the LFDA to original high dimensional descriptors, one problem is the computation cost. Suppose the vector data has a dimension of d, LFDA has to solve the eigenvalue a matrix with dimension $d \times d$. In some descriptors the d could be more than 20000 and thus the computation cost is intractable.

4.1.4 Kernel local fisher discriminant analysis(KLFDA)

KLFDA [35] is the nonlinear version of LFDA. Most dimensionality reduction methods including PCA, LDA and LFDA are linear dimensionality reduction methods. However, when descriptors data are non-linear in feature space, its hard to capture its between-class discriminant information with linear reduction methods. One alternative method is to nonlinearly map input descriptors $\boldsymbol{x}_i$ to higher dimensional feature space Φ by a function $\phi(\boldsymbol{x}_i)$, again the LFDA is performed in feature space Φ. Thus the transformation matrix T can be computed by equation

$$\boldsymbol{T} = \arg\min \frac{\boldsymbol{T}^T \boldsymbol{S}_\phi^{(b)} \boldsymbol{T}}{\boldsymbol{T}^T \boldsymbol{S}_\phi^{(w)} \boldsymbol{T}} \tag{4.14}$$

where $\boldsymbol{S}_\phi^{(b)}$ and $\boldsymbol{S}_\phi^{(w)}$ is the between class scatter and within class scatter in mapped feature space Φ.

Note that the transformation matrix $\boldsymbol{T} \in \Phi$, it's computationally expensive to explicitly compute the mapping function ϕ and perform LFDA in feature space Φ because the dimension of Φ may be infinite. Rather than explicitly computing, the mapping function ϕ can be implicit and the feature space Φ can be defined by the inner product of features in Φ. Kernel trick [] is used here and a kernel function can be defined as the inner product of mapped vectors $\phi(\boldsymbol{x}_i)$ and $\phi(\boldsymbol{x}_j)$ by equation

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) =< \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) >, \tag{4.15}$$

the $< \cdot >$ is the inner product. There are many kinds of kernel like linear kernel, polynomial kernel and radial basis function (RBF) kernel. In this paper the RBF kernel is adopted. A RBF kernel is defined as

$$k_{RBF}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp^{(-\gamma||\boldsymbol{x}_i - \boldsymbol{x}_j||^2)} . \tag{4.16}$$

Suppose $\boldsymbol{X}$ is the sample descriptors matrix, and we have

$$\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n), \tag{4.17}$$

and the label vector is $\boldsymbol{l} = (l_1, l_2, \cdots, l_n)$. Then the kernel matrix of $\boldsymbol{X}$ can be computed as following equation:

$$\boldsymbol{K} = \phi(\boldsymbol{X})^T \phi(\boldsymbol{X}) \tag{4.18}$$

and we have

$$\boldsymbol{K}_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j) =< \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) >= \exp^{(-\gamma||\boldsymbol{x}_i - \boldsymbol{x}_j||^2)} \tag{4.19}$$

In [19] the authors proposed fast computation of LFDA by replacing $\boldsymbol{S}^{(b)}$ with the local scatter mixture matrix $\boldsymbol{S}^{(m)}$defined by

$$\begin{aligned}\boldsymbol{S}^{(m)} &= \boldsymbol{S}^{(b)} + \boldsymbol{S}^{(w)} \\ \boldsymbol{S}^{(m)} &= \frac{1}{2}\sum_{i,j=1} \boldsymbol{A}_{i,j}^{(m)}(\boldsymbol{x}_i - (\boldsymbol{x}_j)(\boldsymbol{x}_i - (\boldsymbol{x}_j)^T\end{aligned} \tag{4.20}$$

and

$$\boldsymbol{A}_{i,j}^{(m)} = \boldsymbol{A}_{i,j}^{(w)} + \boldsymbol{A}_{i,j}^{(w)} \tag{4.21}$$

according to indentify(Fukunaga, 1990)

$$tr((\boldsymbol{T}^T\boldsymbol{S}^{(w)}\boldsymbol{T})^{(-1)}(\boldsymbol{T}^T\boldsymbol{S}^{(m)}\boldsymbol{T}) = tr((\boldsymbol{T}^T\boldsymbol{S}^{(w)}\boldsymbol{T})^{(-1)}(\boldsymbol{T}^T\boldsymbol{S}^{(b)}\boldsymbol{T}) + m \tag{4.22}$$

equation 4.1.3 is equal to

$$\boldsymbol{T}_{LFDA} = \arg\min_{\boldsymbol{T}}(\frac{\boldsymbol{T}^T\boldsymbol{S}^{(m)}\boldsymbol{T}}{\boldsymbol{T}^T\boldsymbol{S}^{(w)}\boldsymbol{T}}) \tag{4.23}$$

and it can be transformed into a eigenvalue decomposition problem

$$\boldsymbol{S}^{(m)}\boldsymbol{t}_i = \lambda\boldsymbol{S}^{(w)}\boldsymbol{t}_i \tag{4.24}$$

Also with the replacement of $\boldsymbol{S}^{(m)}$, in [19] the author summarized that

$$\boldsymbol{S}^{(m)} = \boldsymbol{X}\boldsymbol{L}^{(m)}\boldsymbol{X}^T \tag{4.25}$$

where $\boldsymbol{L}^{(m)} = \boldsymbol{D}^{(m)} - \boldsymbol{A}^{(m)}$, and $\boldsymbol{D}^{i,i} = \sum_{j=1}^{n} \boldsymbol{A}^{(m)}$. Also $\boldsymbol{S}^{(w)}$ can be represented as

$$\boldsymbol{S}^{(w)} = \boldsymbol{X}\boldsymbol{L}^{(w)}\boldsymbol{X}^T \tag{4.26}$$

where $\boldsymbol{L}^{(w)} = \boldsymbol{D}^{(w)} - \boldsymbol{A}^{(m)}$, and $\boldsymbol{D}^{i,i} = \sum_{j=1}^{n} \boldsymbol{A}^{(w)}$. Therefore, equation 4.1.4 can be represented as

$$\boldsymbol{X}\boldsymbol{L}^{(m)}\boldsymbol{X}^T\boldsymbol{t}_i = \lambda\boldsymbol{X}\boldsymbol{L}^{(w)}\boldsymbol{X}^T\boldsymbol{t}_i \tag{4.27}$$

the eigen vector $\boldsymbol{t}_i$ can be represented as $\boldsymbol{t}_i = \boldsymbol{X}\gamma, vector\gamma_i \in R^n$, with this replacement, we left multiply $\boldsymbol{X}^T$ to equation 4.1.4 to get

$$\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{L}^{(m)}\boldsymbol{X}^T\boldsymbol{X}\gamma_i = \lambda\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{L}^{(w)}\boldsymbol{X}^T\boldsymbol{X}\gamma_i \tag{4.28}$$

and by the kernel trick, its represented as

$$\boldsymbol{K}\boldsymbol{L}^{(m)}\boldsymbol{K}\gamma_i = \lambda\boldsymbol{K}\boldsymbol{L}^{(w)}\boldsymbol{K}\gamma_i \tag{4.29}$$

One example of using KLFDA to reduce dimension and classify the nonlinear data clusters can be shown in figure 4.1.4, 4.1.4 and 4.1.4. Three classes with five clusters are distributed on a 2-D plane, by KLFDA dimension reduction its 1-D dimension reduced data distribution are shown in figure 4.1.4 and figure 4.1.4. It shows that for those clusters the gaussian kernel are better than linear kernel because the dimensional reduced data are more separate when using gaussian kernel function.

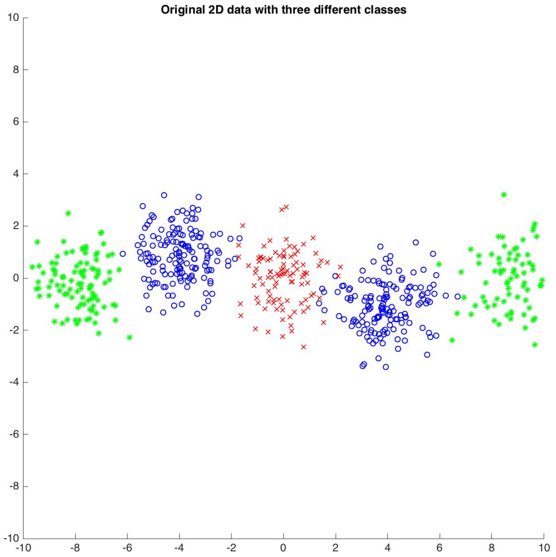


FIGURE 4.1: Example of five clusters belong to three classes

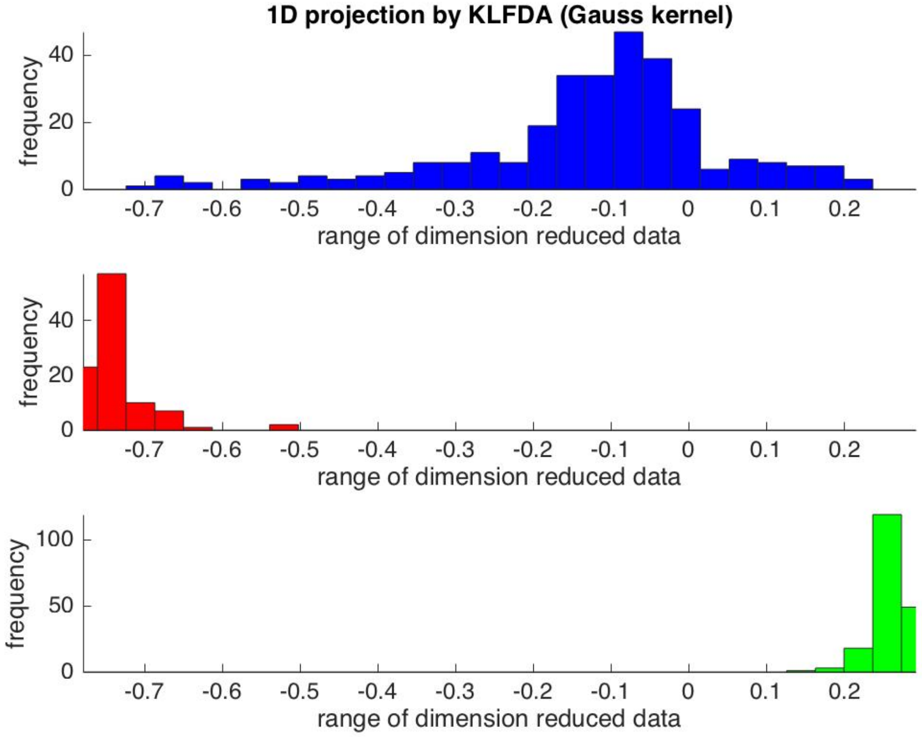


FIGURE 4.2: 1-D distribution of dimension reduced data with gaussian kernel

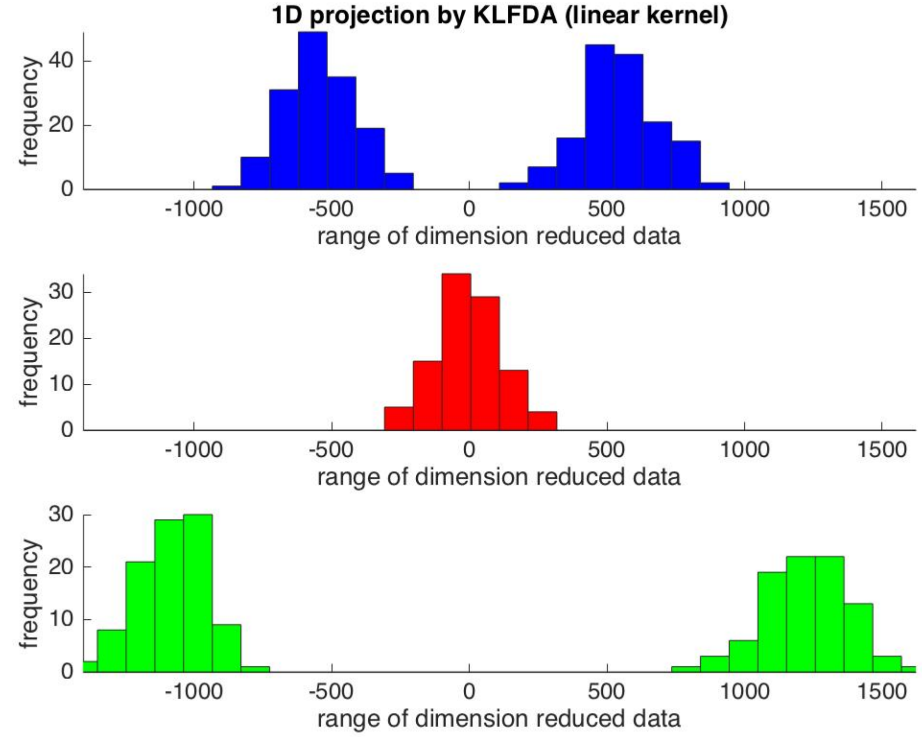


FIGURE 4.3: 1-D distribution of dimension reduced data with linear kernel

4.2 Mahalanobis distance

The Mahalanobis distance [32] based metric learning has received much attention in similarity computing. The Mahanalobis distance of two observations $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as

$$D(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T \boldsymbol{M}(\boldsymbol{x} - \boldsymbol{y}), \tag{4.30}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are $d \times 1$ observation vectors, $\boldsymbol{M}$ is a semi-positive definite matrix. Since $\boldsymbol{M}$ is semi-positive definite matrix, $\boldsymbol{M}$ can be decomposed as $\boldsymbol{M} = \boldsymbol{W}^T\boldsymbol{W}$, and Mahanalobis distance can also be written as

$$D(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T \boldsymbol{W}^T \boldsymbol{W}(\boldsymbol{x} - \boldsymbol{y}) = ||\boldsymbol{W}(\boldsymbol{x} - \boldsymbol{y})|| \tag{4.31}$$

Therefore, Mahanalobis distance can be regarded as a variant of Euclidean distance.

4.3 Gradient descent optimization

Given a multivariate function $F(\boldsymbol{x})$, $\boldsymbol{x}$ is a d dimensional vector, if $f(\boldsymbol{x})$ is continuous and differentiable in the neighbour of point $\boldsymbol{x}$ for all $\boldsymbol{x}$, then $f(\boldsymbol{x})$ decreases fastest in the direction of negative gradient of F at $\boldsymbol{x}$. To compute the minimum of $F(\boldsymbol{x})$, an iterative method can be used by updating F with respect to $\boldsymbol{x}$. If the updating step λ is small enough, by updating $\boldsymbol{x}$ with

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \lambda \boldsymbol{G} \tag{4.32}$$

we have

$$F(\boldsymbol{x}_{t+1}) \geq F(\boldsymbol{x}_t). \tag{4.33}$$

This process is repeated until certain condition is met, generally when gradient $||\boldsymbol{G}|| \leq \eta$, η is a very small positive integer.

Analysis of steepest gradient descent method The advantages of gradient descent are it's always downhill and it can avoid the saddle points. Besides, it's very efficient when initial value of $F(\boldsymbol{x})$ is further from minimum. However, there are a few shortcomings of gradient descent method. The first one is the convergence value of gradient descent might be the local minima of $F(\boldsymbol{x})$ if $F(\boldsymbol{x})$ is not monotonic 4.5. In this case the convergence value will depend on the initial value of $\boldsymbol{x}$.

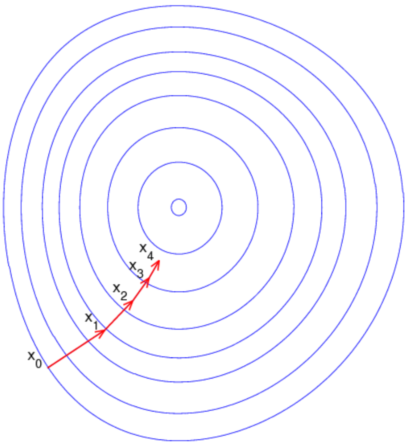


FIGURE 4.4: Steepest gradient descent

Another shortcoming is the converging speed goes very slow when approaching the minimum. One example of slow approaching speed is the zigzag approaching case in figure 4.6. The third shortcoming is linear search in gradient descent might cause some problem.

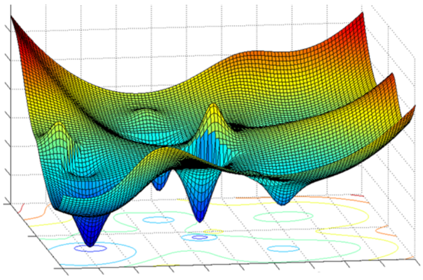


FIGURE 4.5: Function with multi local minimums

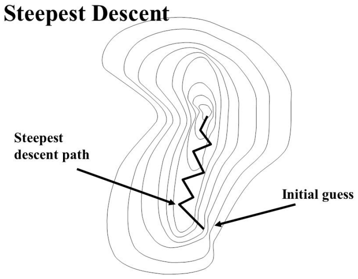


FIGURE 4.6: Zigzagging downhill valley

4.4 Metric learning based on sample pairs distance comparison

Inspired by [42], in this paper, a similar metric learning based on iteration computation is used. For a sample descriptor $\boldsymbol{x}_i$, its positive pairwise set is defined as $\{\boldsymbol{x}_i, \boldsymbol{x}_j\}$, where class ID $y_i = y_j$. Also the negative pairwise set can be defined as $\{\boldsymbol{x}_i, \boldsymbol{x}_j\}$, where $y_i \neq y_j$. Similar with [45], this method is also based on similarity comparison. The difference is in [45], for all possible positive and negative pairs, the distance between positive pairs must be smaller than the distance between negative pairs. Since it has to compare all possible positive and negative pairs, its computation complexity is quite expensive. To decrease complexity, a simplified version is proposed as the top-push distance metric learning [42]. Since re-identification is a problem of ranking, it is desired that the rank-1 descriptor should be the right match. Given a Mahanalobis matrix $\boldsymbol{M}$, for samples $\boldsymbol{x}_i, i = 1, 2, 3, \cdots, n$, n is the number of all samples, the requirement is distance between positive pair should be at least 1 unit smaller than the minimum distance of all negative pair. This can be denoted as

$$D(\boldsymbol{x}_i, \boldsymbol{x}_j) + \rho < \min D(\boldsymbol{x}_i, \boldsymbol{x}_k), y_i = y_j, y_i \neq y_k. \tag{4.34}$$

ρ is a slack variable and $\rho \in [0, 1]$. This equation can be transformed into a optimization problem as

$$\min \sum_{y_i = y_j} \max\{D(\boldsymbol{x}_i, \boldsymbol{x}_j) - \min_{y_i \neq y_k} D(\boldsymbol{x}_i, \boldsymbol{x}_k) + \rho, 0\}. \tag{4.35}$$

However, the equation above only penalizes the minimum interclass distance. Another term is needed to penalize large intraclass distance. That is, the sum of intraclass distance should be as small as possible. This term is denoted as

$$\min \sum_{y_i = y_j} D(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{4.36}$$

To combine equations above, a ratio factor α is assigned to equation (4.35) and (4.36) so that the target function can be denoted as

$$\begin{split} f(\boldsymbol{M}) = (1 - \alpha) \sum_{\boldsymbol{x}_i, x_j, \boldsymbol{y}_i = y_j} D(\boldsymbol{x}_i, \boldsymbol{x}_j) + \\ \alpha \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j, y_i = y_j} \max\{D(\boldsymbol{x}_i, \boldsymbol{x}_j) - \min_{y_i \neq y_k} D(\boldsymbol{x}_i, \boldsymbol{x}_k) + \rho, 0\} \end{split} \tag{4.37}$$

In this way the problem is transformed to an optimization problem. Notice that equation 16 can be denoted as

$$D(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T \boldsymbol{M} (\boldsymbol{x} - \boldsymbol{y}) = trace(\boldsymbol{M}\boldsymbol{X}_{i,j}) \tag{4.38}$$

where $\boldsymbol{X}_{i,j} = \boldsymbol{x}_i * \boldsymbol{x}_j^T$, and $trace$ is to compute matrix trace. Therefore, equation 21 can be transformed as follow,

$$\begin{aligned} f(\boldsymbol{M}) = (1-\alpha) \sum_{y_i=y_j} trace(\boldsymbol{M}\boldsymbol{X}_{i,j}) \\ +\alpha \sum_{y_i=y_j, y_i \neq y_k} \max\{trace(\boldsymbol{M}\boldsymbol{X}_{i,j}) - trace(\boldsymbol{M}\boldsymbol{X}_{i,k}) + \rho, 0\} \end{aligned} \tag{4.39}$$

To minimize equation 23, the gradient descent method is used. The gradient respect to $\boldsymbol{M}$ is computed as

$$\boldsymbol{G} = \frac{\partial f}{\partial \boldsymbol{M}} = (1-\alpha) \sum_{y_i=y_j} \boldsymbol{X}_{i,j} + \alpha \sum_{y_i=y_j, y_i \neq y_k} (\boldsymbol{X}_{i,j} - \boldsymbol{X}_{i,k}) \tag{4.40}$$

The iteration process can be summarized as in table 4.4

TABLE 4.1: Optimization algorithm of Mahanalobis distance matrix learning

```
Gradient optimization algorithm for target function
Input Descriptors of training person pairs
Output A SPD matrix
Initialization
Initialize $\boldsymbol{M}_0$ with eye matrix $\boldsymbol{I}$;
Compute the initial target function value $f_0$ with $\boldsymbol{M}_0$;
Iteration count $t = 0$;
while(not converge)
Update $t = t + 1$;
Find $\boldsymbol{x}_k$ for all sample points $\boldsymbol{x}_i$, where $y_i \neq y_k$;
Update gradient $\boldsymbol{G}_{t+1}$ with equation 12;
Update $\boldsymbol{M}$ with equation : $\boldsymbol{M}_{t+1} = \boldsymbol{M}_t - \lambda \boldsymbol{G}_t$;
Project $\boldsymbol{M}_{t+1}$ to the positive semi-positive definite space;
Update the target value $f|_{\boldsymbol{M}=\boldsymbol{M}_{t+1}}$;
end while
return $\boldsymbol{M}$
```

In each iteration, to make sure the updated $\boldsymbol{M}$ is a SPD matrix, first a eigenvalue decomposition is performed on $\boldsymbol{M}$, and we have

$$\boldsymbol{M} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^T \tag{4.41}$$

here Λ is a diagonal matrix with diagonal elements are eigenvalues. Then the negative eigenvalues in $\boldsymbol{V}$ are removed and the corresponding eigenvectors in $\boldsymbol{V}$ are also removed. Then $\boldsymbol{M}$ is restored by equation (4.41).

Chapter 5

Experiment Settings

5.1 Datasets and evaluation settings

VIPeR dataset is the most used dataset in person re-identification. In this dataset there are 632 different individuals and for each person there are two outdoor images from different viewpoints. All the images are scaled into 48×128. In this experiment the we randomly select 316 individuals from cam a and cam b as the training set, the rest images in cam a are used as probe images and those in cam b as gallery images. This process is repeated 10 times to reduce error.
CUHK1 dataset contains 971 identities from two disjoint camera views. The cameras are static in each pair of view and images are listed in the same order. For each individual, there are two images in each view. All images are scaled into 60×160. In this paper, we randomly select 485 image pairs as training data and the rest person pairs are used for test data.

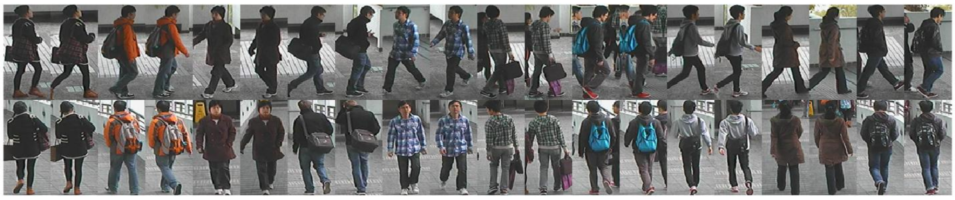


FIGURE 5.1: Pedestrians in prid_450 dataset

Prid_2011 dataset consists of images extracted from multiple person trajectories recorded from two different, static surveillance cameras. Images from these cameras contain a viewpoint change and a stark difference in illumination, background and camera characteristics.Camera view A shows 385 persons, camera view B shows 749 persons. The first 200 persons appear in both camera views, The remaining persons in each camera view complete the gallery set of the corresponding view. Hence, a typical evaluation consists of searching the 200 first persons of one camera view in all persons of the other view. This means that there are two possible evaluation procedures, either the probe set is drawn from view A and the gallery set is drawn from view B. In this paper, we randomly select 100 persons that appeared in both camera views as training pairs, and the remaining 100 persons of the 200 person pairs from camera a is used as probe set while the 649 remaining persons from camera B are used for gallery images.

Prid_450s dataset contains 450 image pairs recorded from two different, static surveillance cameras. Additionally, the dataset also provides an automatically generated, motion based foreground/background segmentation as well as a manual segmentation of parts of a person. The images are stored in two folders that represent the two camera views. Besides the original images , the folders also contain binary masks obtained from motion segmentation, and manually segmented masks. In this test, we randomly select 225 persons from each of two camera views as the training set, and the remaining persons are left as gallery and probe images.

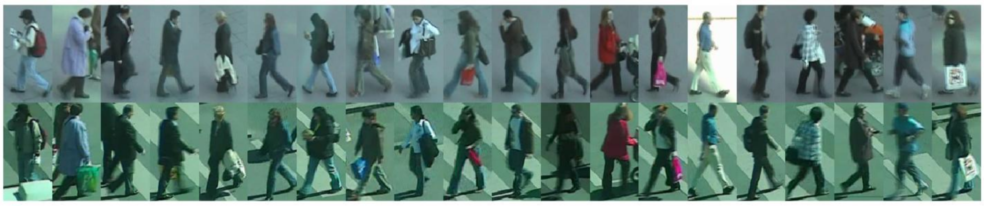


FIGURE 5.2: Pedestrians in prid_450s dataset

GRID There are two camera views in this dataset. Folder probe contains 250 probe images captured in one view (file names starts from 0001 to 0250). Folder gallery contains 250 true match images of the probes (file names starts from 0001 to 0250). Besides, in gallery folder there are a total of 775 additional images that do not belong to any of the probes (file name starts with 0000). These extra images should be treated as a fixed portion in the testing set during cross validation. In this paper, we randomly select 125 persons from those 250 persons appeared in both camera views as training pairs, and the remaining persons in probe folder is used as probe images while the remaining 125 persons and those 775 additional persons from gallery folder are used as gallery images.

FIGURE 5.3: Pedestrians in GRID dataset

TABLE 5.1: Testing setting for different datasets

Dataset	training	probe	gallery	cam_a	cam_b
VIPeR	316	316	316	632	632
CUHK1	485	486	486	971	971
PRID_2011	100	100	649	385	749
PRID_450s	225	225	225	450	450
GRID	125	125	900	250	1025

5.2 The influence of mean removal and L_2 normalization

In [22], mean removal and L_2 normalization is found to improve performance by 5.1%. The reason for this is mean removal and normalization can reduce the impact of extremas of descriptors. When testing proposed metric learning, we find the mean removal can slightly improve performance. A comparison between performance of original descriptors and preprocessed descriptors is shown in Tables 5.2, 5.2, 5.2, 5.2, 5.2, all those datasets are tested by proposed metric. The original GOG means no mean removal and normalization. It shows that the mean removal and normalization has a slight improvement around 0.5% on the performance on all five datasets. Since preprocessing are required to test XQDA, the mean removal and normalization are operated on descriptors in this experiment.

TABLE 5.2: The influence of data preprocessing on VIPeR

	Rank(%)				
Terms	1	5	10	15	20
Original GOG	43.01	74.91	84.87	89.81	93.32
Preprocessed GOGrgb	43.77	74.84	85.25	90.32	93.89
Original GOGfusion	48.77	77.47	87.41	91.52	94.27
Preprocessed GOGfusion	48.32	76.90	87.78	91.93	94.49

TABLE 5.3: The influence of data preprocessing on CUHK1

	Rank(%)				
Terms	1	5	10	15	20
Original GOGrgb	56.11	83.77	90.10	92.65	94.28
Preprocessed GOGrgb	55.91	84.24	90.41	93.15	94.67
Original GOGfusion	57.10	84.65	90.35	92.88	94.65
Preprocessed GOGfusion	56.67	84.49	90.51	93.31	94.84

TABLE 5.4: The influence of data preprocessing on prid_2011

	Rank(%)				
Terms	1	5	10	15	20
Original GOGrgb	24.80	52.10	63.20	69.90	72.90
Preprocessed GOGrgb	23.80	52.20	63.50	70.20	73.50
Original GOGfusion	32.20	56.60	67.00	73.10	77.70
Preprocessed GOGfusion	32.30	57.40	66.30	73.40	78.00

TABLE 5.5: The influence of data preprocessing on prid_450s

	Rank(%)				
Terms	1	5	10	15	20
Original GOGrgb	60.93	84.31	91.29	94.00	96.18
Preprocessed GOGrgb	60.71	84.53	91.29	94.13	96.27
Original GOGfusion	63.07	86.67	92.53	95.20	96.98
Preprocessed GOGfusion	62.80	86.58	92.36	95.29	96.89

TABLE 5.6: The influence of data preprocessing on GRID

	Rank(%)				
Terms	1	5	10	15	20
Original GOGrgb	22.96	41.92	51.68	58.72	64.64
Preprocessed GOGrgb	22.64	43.68	52.00	59.04	65.04
Original GOGfusion	24.32	44.56	54.80	62.40	66.64
Preprocessed GOGfusion	23.92	44.64	54.88	62.32	66.40

5.3 Parameters setting

In this experiment, there are a few parameters for the iteration computing including slack variable ρ, maximal iteration T, gradient step λ, the interclass and intraclass limitation factor α and the updating ratio β. Firstly the slack variable ρ is initialized as 1 to ensure the minimum interclass distance is 1 larger than intraclass distance at least. The step size of gradient updating λ is initialized as 0.01. When target value f increases, λ is scaled by a factor 0.5, and λ is scaled by 1.01 when target value f decreases. To judge if target value converges, the thresh β is defined as the ratio target value change versus previous target value, that is, $\beta = \frac{(f_{t+1}-f_t)}{f_t}$. According many experiment trials, when it satisfies $\beta = 10^{-5}$, the target value converges and the iteration is stopped. The maximal iteration times is set to 100 since the target value f will converge in around 15 iterations. The last parameter for the iteration is α, to know the best value for α, we tried 11 different values ranges from 0 to 1 with a step of 0.1, and find that the rank 1 and rank 5 scores reach maxima at interval $[0.7, 0.8]$. Then another ten trials with alpha ranging from $[0.7, 0.8]$ with a step of 0.01. The best α value should have as large top rank scores as possible and at last we find that the optimal value for α is 0.76. A form of all parameters are shown in Form 7.

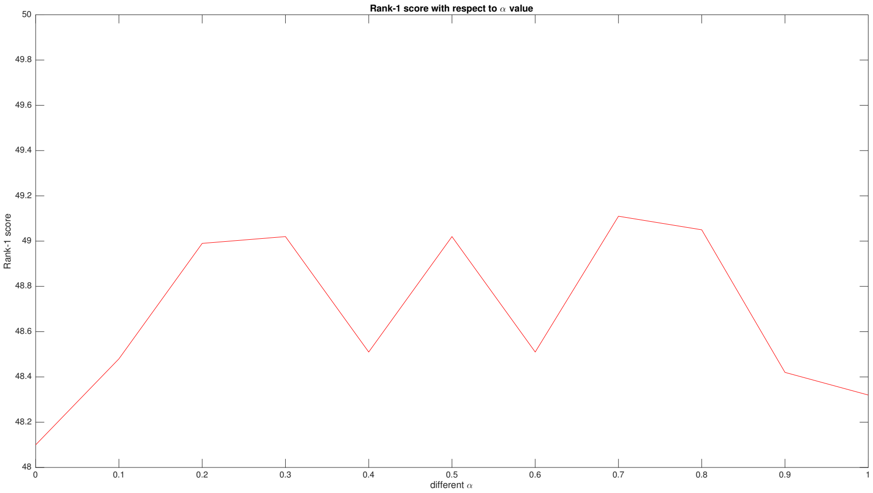


FIGURE 5.4: Rank 1 scores with respect to α on VIPeR

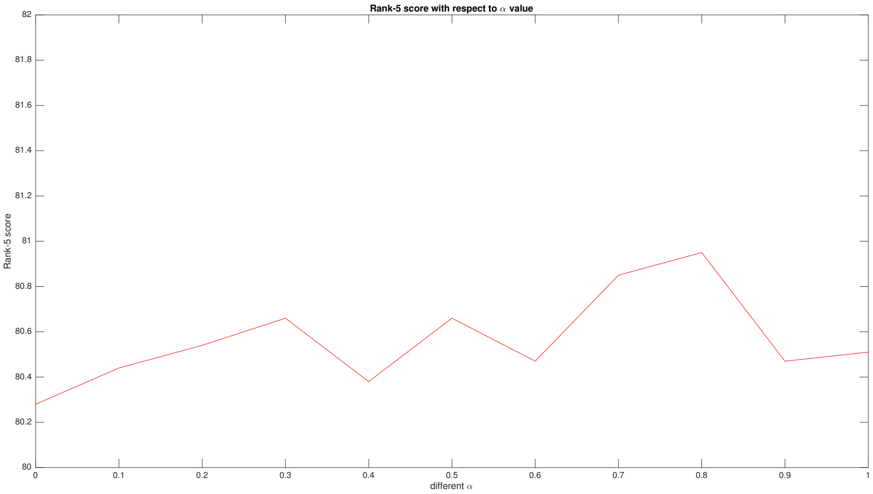


FIGURE 5.5: Rank 5 scores with respect to α on VIPeR

TABLE 5.7: Parameters setting

Paramters	α	Thresh	Step	Max iteration	Slack variable
Values	0.76	10^{-5}	0.01	100	1

Performance measuring The cumulative matching curve is used to measure the descriptor performance. The score means the probability that the right match is within the top n samples. A better CMC curve is expected to have a high rank-1 value and reaches 1 as fast as possible.

5.4 Results and experiment analysis

In this paper, we compare proposed metric with other state-of-the-art metrics including NFST [43], XQDA [18]. NFST is a metric which learn a null space for descriptors so that the the same class descriptors will be projected to a single point to minimize within class scatter matrix while different classes are projected to different points. This metric is a good solution to small sample problems in person re-identification. XQDA is quite similar with many other metrics, which learns a projection matrix W and then a Mahanalobis SPD matrix $\boldsymbol{M}$ is learned in the subspace. Those two metric are proved to have state-of-the-art performance with many other methods. The GOGrgb in all forms stands for the hierarchical gaussian descriptor in RGB color space while GOGfusion stands for the one in four different color spaces {RGB, Lab, HSV, nRnR}.

VIPeR A comparison form is given in Table 3. Some of recent results are also included in this form. We can find that the rank scores are better than those of NFST and XQDA in terms of both GOGrgb and GOGfusion. More specifically, the rank 1, rank 5, rank 10, rank 15 and rank 20 scores of proposed metric learning are 0.76%, 0.92%, 1.39%, 1.08%, 1.52% higher than those of GOGrgb + XQDA, and the rank 1, rank 5, rank 10, rank 15 and rank 20 GOGfusion scores of proposed metric learning are 0.35%, -0.54%, 0.98%, 0.66%, 0.79% higher than GOGfusion + XQDA respectively. Also we can see that the proposed metric learning has a better performance than NFST.

TABLE 5.8: Performance of different metrics on VIPeR

	Rank(%)				
Methods	1	5	10	15	20
GOGrgb+NFST	43.23	73.16	83.64	89.59	92.88
GOGrgb+XQDA	43.01	73.92	83.86	89.24	92.37
GOGrgb+Proposed	43.77	74.84	85.25	90.32	93.89
GOGfusion+NFST	47.15	76.39	87.31	91.74	94.49
GOGfusion+XQDA	47.97	77.44	86.80	91.27	93.70
GOGfusion+Proposed	48.32	76.90	87.78	91.93	94.49

CUHK1 We can find that the rank 1, rank5, rank 10, rank 15, rank 20 score

of GOGrgb combined with proposed metric are 5.4%, 4.18%,3.31%,2.16%,1.46% higher than XQDA, and 0.31%,1.22%,1.34%,1.17%, 1.11% than NFST. Also the rank 1, rank5, rank 10, rank 15, rank 20 score of GOGfusion combined with proposed metric are 4.57%, 2.64%, 0.70%, 1.33%, 0.83% higher than GOGfusion combined with XQDA, and 0.41%, 0.83%, 0.88%, 1.09%, 1.14% than GOGfusion combined with NFST.

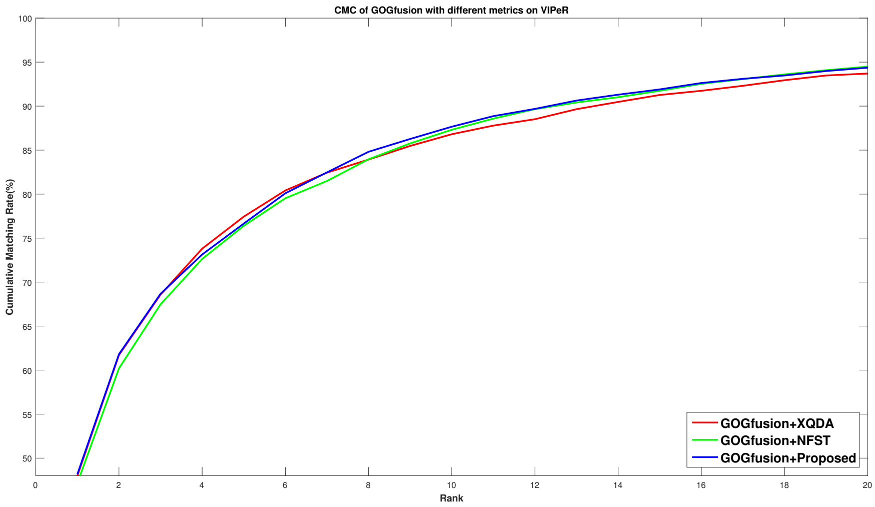


FIGURE 5.6: CMC curves on VIPeR comparing different metric learning

TABLE 5.9: Performance of different metrics on CUHK1

	Rank(%)				
Methods	1	5	10	15	20
GOGrgb+NFST	55.60	83.02	89.07	91.98	93.56
GOGrgb+XQDA	50.51	80.06	87.10	90.99	93.21
GOGrgb+Proposed	55.91	84.24	90.41	93.15	94.67
GOGfusion+NFST	56.26	83.66	89.63	92.22	93.70
GOGfusion+XQDA	52.10	81.85	88.81	91.98	94.01
GOGfusion+Proposed	56.67	84.49	90.51	93.31	94.84

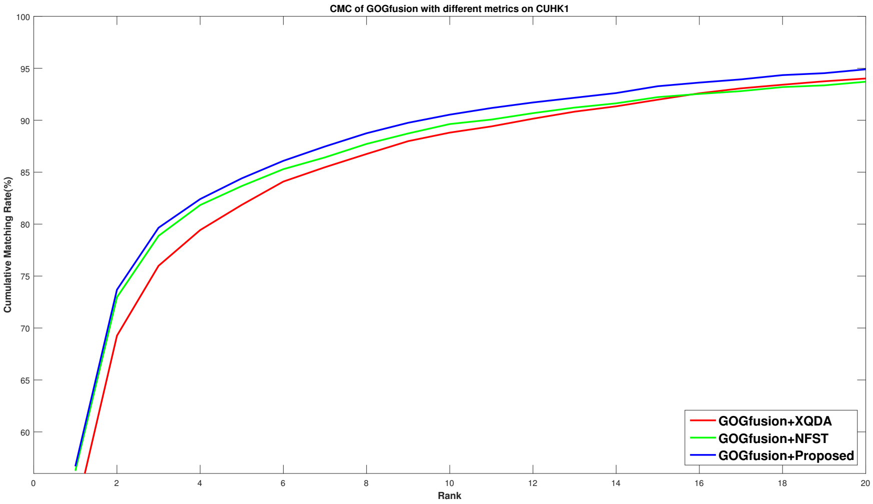


FIGURE 5.7: CMC curves on CUHK1 comparing different metric learning

TABLE 5.10: Performance of different metrics on prid_2011

	Rank(%)				
Methods	1	5	10	15	20
GOGrgb+NFST	26.60	53.80	62.90	71.30	75.40
GOGrgb+XQDA	31.10	55.70	66.10	72.40	76.10
GOGrgb+Proposed	23.80	52.20	63.50	70.20	73.50
GOGfusion+NFST	34.10	58.30	67.60	73.80	78.30
GOGfusion+XQDA	38.40	61.30	70.80	75.60	79.30
GOGfusion+Proposed	32.30	57.40	66.30	73.40	78.00

Prid_2011 The rank 1, rank5, rank 10, rank 15, rank 20 score of GOGfusion combined with proposed metric are 6.1%, 3.9%, 4.5%, 2.2% and 1.3% lower than GOGfusion combined with XQDA. The performance of NFST is slightly better than proposed metric. Also in terms of GOGrgb XQDA and NFST has better performance than the proposed one. So in this dataset the proposed metric has worse performance than XQDA and NFST.

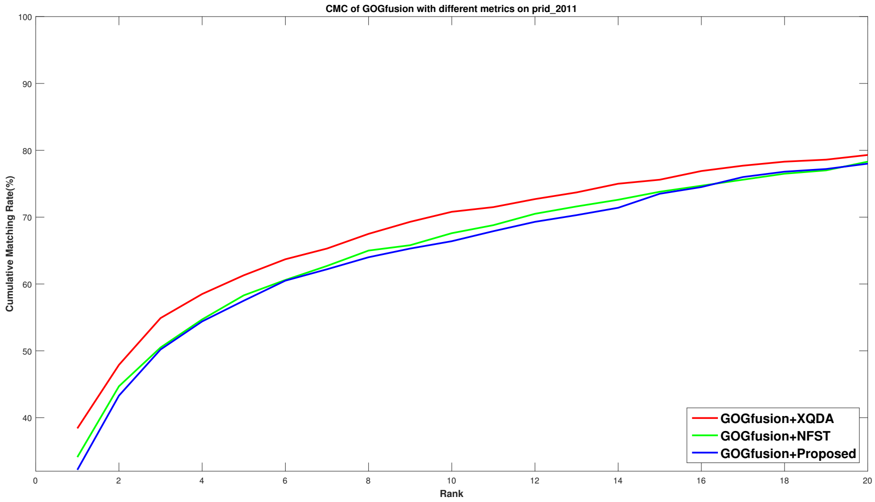


FIGURE 5.8: CMC curves on prid_2011 comparing different metric learning

TABLE 5.11: Performance of different metrics on prid_450s

	Rank(%)				
Methods	1	5	10	15	20
GOGrgb+NFST	61.96	84.98	90.53	94.09	96.09
GOGrgb+XQDA	65.29	85.02	91.13	94.76	96.49
GOGrgb+Proposed	60.71	84.53	91.29	94.13	96.27
GOGfusion+NFST	64.53	86.62	92.93	95.78	97.42
GOGfusion+XQDA	68.40	87.42	93.47	95.69	97.02
GOGfusion+Proposed	62.80	86.58	92.36	95.29	96.89

Prid_450s In this dataset, we can find the rank 1 score of XQDA and NFST is higher than proposed metric, but they have almost the same rank 5, rank 10, rank 15, and rank 20 scores with respect to both kinds of descriptors.

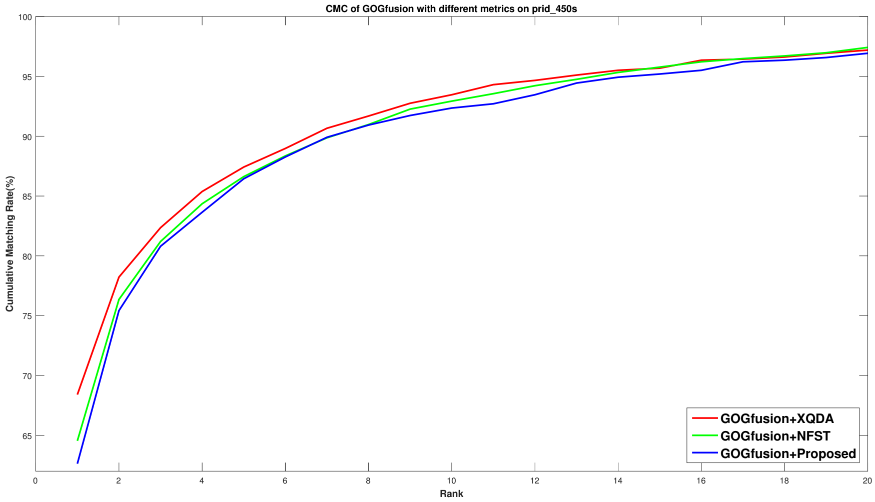


FIGURE 5.9: CMC curves on prid_450s comparing different metric learning

TABLE 5.12: Performance of different metrics on GRID

	Rank(%)				
Methods	1	5	10	15	20
GOGrgb+NFST	21.84	41.28	50.96	57.44	62.88
GOGrgb+XQDA	22.64	43.92	55.12	61.12	66.56
GOGrgb+Proposed	22.64	43.68	52.00	59.04	65.04
GOGfusion+NFST	23.04	44.40	54.40	61.84	66.56
GOGfusion+XQDA	23.68	47.28	58.40	65.84	69.68
GOGfusion+Proposed	23.92	44.64	54.88	62.32	66.40

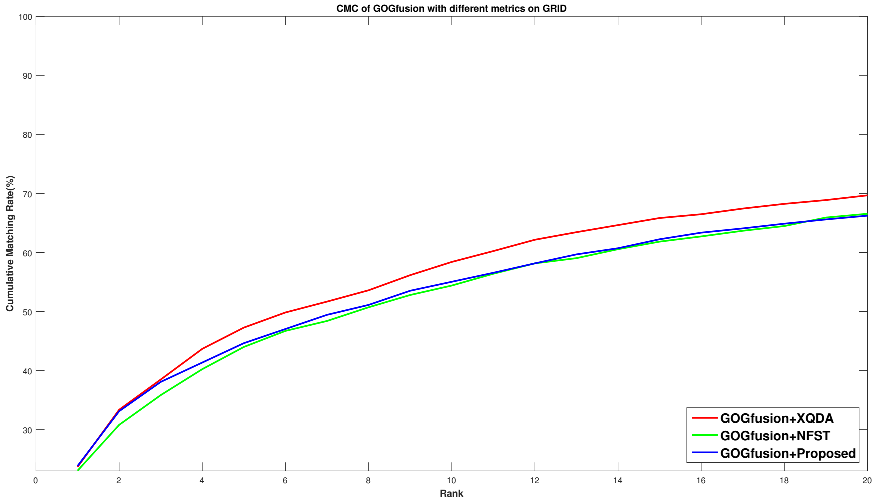


FIGURE 5.10: CMC curves on GRID comparing different metric learning

GRID We can see that the rank 1 score of proposed metric are 0.24% higher than XQDA and 0.88% higher than NFST in terms of GOGfusion, but XQDA outperforms proposed metric on rank 5, rank 10, rank 15 and rank 20 scores. Besides, proposed metric outperforms NFST on rank 5, rank 10, rank 15 scores.

In summary, the Re-ID performance is improved in VIPeR, CUHK01 dataset, and has almost the same performance with NFST and XQDA on prid_450s dataset. Specifically, proposed metric learning has the best rank 1 score in GRID dataset and its performance is only second to XQDA. The proposed metric has superior performance for following reasons: (1) dimension reduction by KLFDA exploits the nonlinearity and the loss of discriminant information between classes are minimized. (2) the simplified relative distance limitation optimization helps to confine the Mahanalobis distance matrix M to discriminate different classes.

Chapter 6

Conclusion

In this paper we combined KLFDA with gradient descent method based metric learning. A semi-positive definite (SPD) matrix is learned on the lower dimension space after dimension reduction by kernel local fisher discriminative analysis. By analysis we can find the proposed metric has better performance than NFST and XQDA on VIPeR and CUHK1 datasets, but XQDA and NFST outperforms the proposed metric learning on Prid_2011 and Prid_450s, and the proposed metric learning has better rank 1 score than NFST and its performance is only second to XQDA on GRID dataset.

Bibliography

[1] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. “Pictorial Structures Revisited: People Detection and Articulated Pose Estimation”. In: (Apr. 2009), pp. 1–8.

[2] Davide Baltieri, Roberto Vezzani, and Rita Cucchiara. “SARC3D: a new 3D body model for People Tracking and Re-identification”. In: (Mar. 2011), pp. 1–10.

[3] Bedagkar-Gala et al. “A Survey of Approaches and Trends in Person Re-identification”. In: *IMAVIS* (Mar. 2014), pp. 1–72.

[4] A Bedagkar-Gala and Shishir K Shah. “Part-based spatio-temporal model for multi-person re-identification”. In: *Pattern Recognition Letters* 33.14 (Oct. 2012), pp. 1908–1915.

[5] Apurva Bedagkar-Gala and Shishir K Shah. “Multiple Person Re-identification using Part based Spatio-Temporal Color Appearance Model”. In: (Oct. 2011), pp. 1–8.

[6] Dapeng Chen et al. “Similarity Learning on an Explicit Polynomial Kernel Feature Map for Person Re-Identification”. In: (Apr. 2015), pp. 1–9.

[7] De Cheng et al. “Person Re-Identification by Multi-Channel Parts-Based CNN With Improved Triplet Loss Function”. In: (July 2016), pp. 1–10.

[8] Shaogang Gong Chen Change Loy Chunxiao Liu and Xinggang Lin. “Person Re-identification: What Features Are Important?” In: (Nov. 2015), pp. 1–11.

[9] Navneet Dalal and Bill Triggs. “Histograms of Oriented Gradients for Human Detection”. In: (Jan. 2017), pp. 1–8.

[10] Jason V Davis et al. “Information-Theoretic Metric Learning”. In: (June 2007), pp. 1–8.

[11] Michael Jones Ejaz Ahmed and Tim K Marks. “An Improved Deep Learning Architecture for Person Re-Identification”. In: (Apr. 2015), pp. 1–9.

[12] M. Farenzena et al. “Person Re-Identification by Symmetry-Driven Accumulation of Local Features”. In: CVPR. Mar. 2016, pp. 1–8.

[13] Pedro F Felzenszwalb and Daniel P Huttenlocher. “Pictorial Structures for Object Recognition”. In: (Jan. 2013), pp. 1–42.

[14] Xiaofei He and Partha Niyogi. “Locality Preserving Projections”. In: (Nov. 2016), pp. 1–8.

[15] Nebojsa Jojic et al. “Stel component analysis: Modeling spatial correlations in image class structure”. In: (Apr. 2009), pp. 1–8.

[16] Arif Khan, Jian Zhang, and Yang Wang. “Appearance-Based Re-identification of People in Video”. In: *2010 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, Nov. 2010, pp. 357–362.

[17] Martin Kostinger et al. “Large Scale Metric Learning from Equivalence Constraints”. In: (Apr. 2012), pp. 1–8.

[18] Shengcai Liao et al. “Person Re-identification by Local Maximal Occurrence Representation and Metric Learning”. In: CVPR. Apr. 2015, pp. 1–10.

[19] *Local Fisher Discriminant Analysis for Supervised Dimensionality Reduction*. Dec. 2016.

[20] Miroslav Lovric, Maung Min-Oo, and Ernst A Ruh. “Multivariate Normal Distributions Parametrized as a Riemannian Symmetric Space”. In: (June 2002), pp. 1–15.

[21] David G Lowe. “Object Recognition from Local Scale-Invariant Features”. In: (June 1999), pp. 1–8.

[22] Tetsu Matsukawa et al. “Hierarchical Gaussian Descriptor for Person Re-Identification”. In: Dec. 2016, pp. 1–10.

[23] Niall McLaughlin, Jesus Martinez del Rincon, and Paul Miller. “Recurrent Convolutional Network for Video-Based Person Re-Identification”. In: (July 2016), pp. 1–10.

[24] Alexis Mignon and Frederic Jurie. “PCCA: A New Approach for Distance Learning from Sparse Pairwise Constraints”. In: (Apr. 2012), pp. 1–7.

[25] Baback Moghaddam, Tony Jebara, and Alex Pentland. “Bayesian Face Recognition”. In: (Jan. 2017), pp. 1–16.

[26] T. Ojala, M. Pietikäinen, and D. Harwood. “A Comparative Study of Texture Measures with Classification Based on Feature Distributions”. In: *Pattern Recognition*. 1996.

[27] T. Ojala, M. Pietikäinen, and D. Harwood. “Performance evaluation of texture measures with classification based on Kullback discrimination of distributions”. In: *ICPR*. 1994.

[28] Takumi Kobayashi Otsu and Nobuyuki. “LNCS 5302 - Image Feature Extraction Using Gradient Local Auto-Correlations”. In: (Aug. 2008), pp. 1–13.

[29] Sakrapee Paisitkriangkrai, Chunhua Shen, and Anton van den Hengel. “Learning to Rank in Person Re-Identification With Metric Ensembles”. In: (Jan. 2015), pp. 1–10.

[30] Sateesh Pedagadi et al. “Local Fisher Discriminant Analysis for Pedestrian Re-identification”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jan. 2013, pp. 3318–3325.

[31] Bryan Prosser et al. “Person Re-Identification by Support Vector Ranking”. In: *British Machine Vision Conference 2010*. British Machine Vision Association, July 2010, pp. 21.1–21.11.

[32] Peter M Roth et al. “Mahalanobis Distance Learning for Person Re-Identification”. In: (July 2014), pp. 1–21.

[33] Riccardo Satta. “Appearance Descriptors for Person Re-identification: a Comprehensive Review”. In: (July 2016), pp. 1–18.

[34] Zhiyuan Shi, Timothy M Hospedales, and Tao Xiang. “Transferring a Semantic Representation for Person Re-Identification and Search”. In: (Apr. 2015), pp. 1–10.

[35] Masashi Sugiyama. “Local Fisher Discriminant Analysis for Supervised Dimensionality Reduction”. In: Dec. 2016.

[36] Oncel Tuzel, Fatih Porikli, and Peter Meer. “Region Covariance: A Fast Descriptor for Detection and Classification”. In: Dec. 2016, pp. 1–14.

[37] Weinberger et al. “Distance Metric Learning for Large Margin Nearest Neighbor Classification”. In: (Feb. 2009), pp. 1–38.

[38] Max Welling. “Kernel Canonical Correlation Analysis”. In: (Mar. 2005), pp. 1–3.

[39] Tong Xiao et al. “Learning Deep Feature Representations with Domain Guided Dropout for Person Re-identification ”. In: (Apr. 2016), pp. 1–10.

[40] Fei Xiong et al. “Person Re-Identification using Kernel-based Metric Learning Methods”. In: (July 2014), pp. 1–16.

[41] Yang Yang et al. “Salient color names for person re-identification”. In: *European Con- ference on Computer Vision (ECCV)*. 2014.

[42] Jinjie You et al. “Top-push Video-based Person Re-identification”. In: (Jan. 2015), pp. 1–9.

[43] Li Zhang, Tao Xiang, and Shaogang Gong. “Learning a Discriminative Null Space for Person Re-identification”. In: (Mar. 2016), pp. 1–10.

[44] Rui Zhao, Wanli Ouyang, and Xiaogang Wang. “Learning Mid-level Filters for Person Re-identification”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, May 2014, pp. 144–151.

[45] WeiShi Zheng, Shaogang Gong, and Tao Xiang. “Person Re-identification by Probabilistic Relative Distance Comparison”. In: (Nov. 2016), pp. 1–8.