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:*
Professor 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 15, 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:

---

"*Thanks to my solid academic training, today I can write hundreds of words on virtually any topic without possessing a shred of information, which is how I got a good job in journalism.*"

Qiangsen He

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 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

# List of Abbreviations

**LAH** **L**ist **A**bbreviations **H**ere

**WSF** **W**hat (it) **S**tands **F**or

# Chapter 1

# Introduction

People re-identification 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 []. 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.

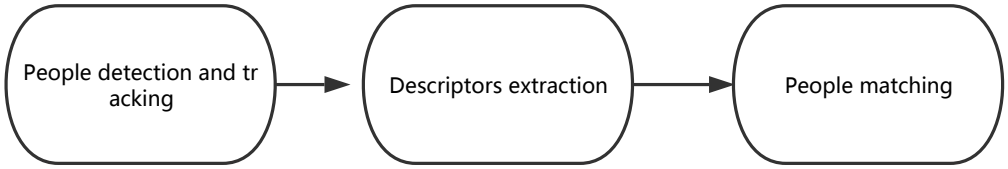


FIGURE 1.1: Re-ID work flow

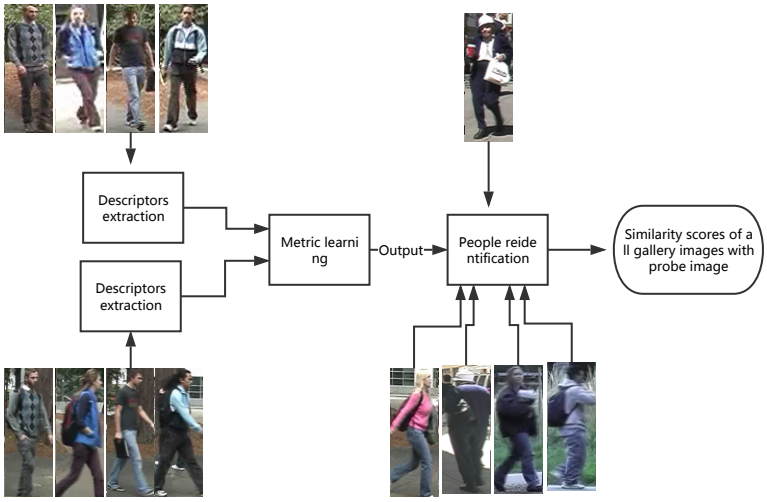


FIGURE 1.2: A typical single shot Re-ID work flow

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 are mostly used to match descriptors like color and texture descriptors. However, though it's straightforward and easy, it's hard to used 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 very popular. In this method a semi-positive defined matrix $\boldsymbol{M}$ is learned while meeting certain limitations. Besides, linear discriminant analysis[] learns a subspace to minimize the within class scatter matrix while maximized inter class scatter matrix. In [] 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** 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[ ], in this dataset, for each person only one image is provides in each camera view. 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 prid2011 dataset which provides a long sequence for each person in a single camera view.

FIGURE 1.3: The VIPeR dataset

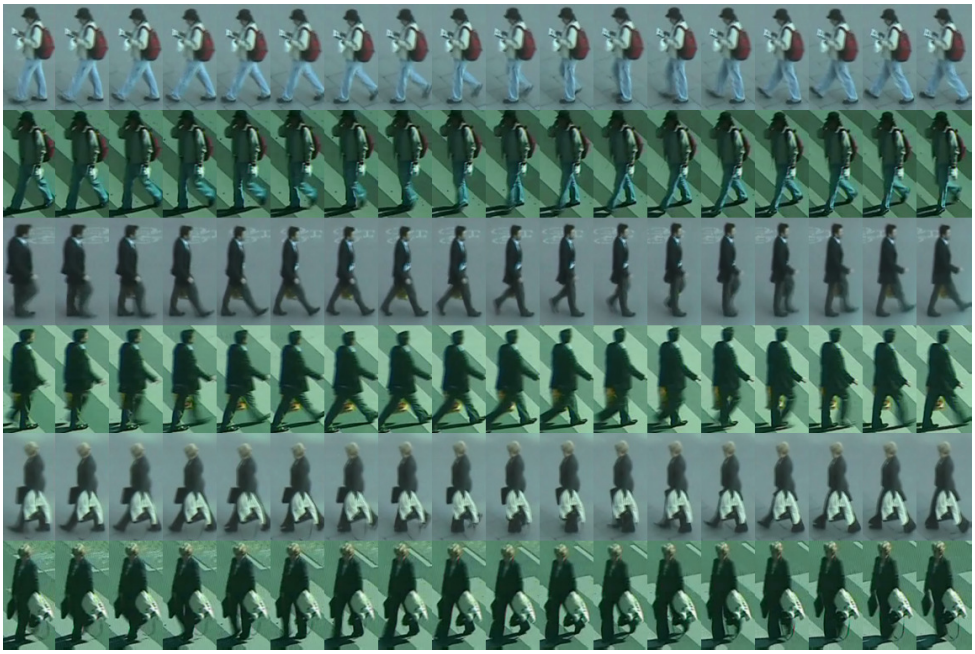

FIGURE 1.4: Samples from prid_2011 dataset

.

## 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 coming processes. 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 [SDALF] is shown is figure [].

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. 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 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 take 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 meet the requirement. It turns out that this metric learning have advance performance when compared with other metric learning methods.

## 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) 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 some descriptors but can decrease the performance of

certain descriptors. 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. This comparison is shown in figure []. 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 bring about more

## 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 comparison. 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

## 2.1 Appearance descriptors

Previous work focus on find more discriminative descriptors and better metric learning. A good descriptor is robust to problems like illumination, low resolution and viewpoint, etc. To model the complex human kinematics, the part-based models are most adopted since human body is non-rigid body. Previous literature mainly contains three kinds of models [1], fixed part models, adaptive part models and the learned part models. The fixed part models are used in [2,3,4], where a silhouette is divided into a fixed number of horizontal and equal stripes, which mainly include head, torso, legs. In [9] the width 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 models vary from one to one according to predefined algorithm. Take [6] 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

$$C(y, \sigma) = \sum d^2(p_i - \hat{p}_i)$$

$$S(y, \sigma) = \sum \frac{1}{W\delta} |A(B[y, y - \delta]) - A(B[y, y + \delta])|$$

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

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

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

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

the axis divides the left and right torso is

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

With those equations above axis can be computed depend on specific image. This method has a relatively high performance.

A work flow of people re-identification is shown in the figure []. Passers-by are detected and tracked with suitable algorithms. We can get bounding boxes containing walking people. The next step is to extract descriptors from the bounding boxes. To decrease the error brought by the background pixels we subtract the background pixels from the bounding boxes. With the silhouettes descriptors can be extracted based on various methods.

With the extracted descriptors re-ID can be summarized as a similarity score computing problem. Classical people re-identification mainly deals with two steps, extracting the individual descriptor and comparing the similarity of different descriptors. Suppose there are two sets, probe set and gallery set . In the probe set , there are a few descriptors representing individuals with unknown ID, while gallery set consists of descriptor with known and labelled IDs. For each image in the probe set we compute its similarity score with every gallery descriptor. Then a list of the similarity scores are listed and the rank-1 descriptor's ID is believed to be probe's ID.

Besides, to model the complex human kinematics, the part-based models are adopted. Previous literature mainly contains three kinds of models[ ],fixed part models, adaptive part models and the learned part models. The fixed part models are used in [ ], where a silhouette is divided into a fixed number of horizontal and equal stripes, which mainly include head, torso, legs. In [ ] the width 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 models vary from one to one according to predefined algorithm. Take [ ] 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 and ,where

The computes the asymmetry of two blobs and computes the difference of two areas. Then the axis between torso and legs are computed as follow,

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

the axis divides the left and right torso is

With those equations above axis can be computed depend on specific image.

The part-based adaptive spatial-temporal model used in [12] 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, that is, statistic classifiers, are trained with manully labelled human body parts images, exploiting features related with edges contained in the images. The pictorial structure is proposed in [4], 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 modeled and deformable configurations are implemented with spring-like connections. This model can quantatively describe visual appearance and model the non-rigid body. In [ ] the body model is made up of N parts and N corresponding part detectors. Suppose be the possible configurations of each part,where is the state of the i-th body part and , and are the coordinates of the part center, is the absolute part orientation, and is the scale size relative to the part size in the training set. Given the image evidence D, the problem is to maximise the posterior probability that the part configuration is correct, and we have , where is the likelihood of image evidence given a particular part configuration and corresponds to a kinematic prior, and those two items can be learned from a training set.

Another example of learned part model is in [15,16], 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 define allowded arrangements of this part with respect to the bounding box. To train each model the Latent Support Vector

Machine is used and in [ 15,16 ] four body parts are detected, namely head, left and right torso and upper legs.

-'-' The implement of every model of different model parts are finished by features. The features can be divided into two categories, the local and global feature. For a specified image or region, to extract the local feature we first divide the whole image into many equal blocks and compute the feature of each block. While the global feature refers to the feature extracted from a whole image or region, and the size of the descriptor is usually fixed.

Global color histogram is a frequently used global feature. For an three-channel image, like RGB image, each channel is quantised into bins separately. The final histogram could be a multi-dimensional or mono-dimensional histogram. For instance, if , for multi-dimensional histogram there will be bins, but if we concatenate the 3 dimensional bins together the dimension can be reduced to bins. Also 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[ ] 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 the scale-extrema of the function difference of Gaussian function with scale , that is,

is the Gaussian filter with standard deviation 1*, is the image, when the image is computed, each point is compared with its neighbors to get extrema.

MSCR (maximally stable color region) is used in [ ]. The MSCR is derived from MSER (maximally stable extreme region), it detects the region with stable color. 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.

RHSP (Recurrent Highly-Structured Patches) is used in [ ]. 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.

——————The discussion starts here Another...

Recently a new 3-D descriptor model is used to represent the individual. Compared with those 2-D models, this model exploits the depth data to help create the descriptor.

For the multi-shots re-ID, there are some additional methods to mention. ——-Convolutional neural network

In [ ] the authors exploit the spatial and temporal information.

The second one is to design the similarity computing methodology. That is, the way to compare how different two descriptors are. Previous method including the Euclidean distance, Bhattacharyya distance and Mahalanobis distance. The Euclidean distance are mostly used for the straightforward descriptors like color and texture descriptors.

Finally, 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 corresponse given a list of computed similarity score, and the first ranked ID is matched as the corresponding ID. For the open-set re-ID problem, re-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.

The part-based adaptive spatial-temporal model used in [7] 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 selects 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, that is, 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 [8], 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 [8] the body model is made up of $N$ parts and $N$ corresponding part detectors. Suppose $L = (l_0, l_1, \ldots, _{N-1})$ be the possible configurations of each part, where $l_i$ is the state of the i-th body part and $l_0 = (x_i, y_i, \theta_i, s_i)$, $x_i$ and $y_i$ are the coordinates of the part center, $\theta_i$ is the absolute part orientation, and $s_i$ is the scale size relative to the part size in the training set. Given the image evidence D, the problem is to maximize the posterior probability $P(D)$ that the part configuration is correct, and we have $P(D) \propto P(L) * P(L)$, where $P(L)$ is the likelihood of image evidence given a particular part configuration and P(L) corresponds to a kinematic prior, and those two items can be learned from a training set.

Another example of learned part model is in [12,13], 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 define allowed arrangements of this part with respect to the bounding box. To train each model the Latent Support Vector Machine is used and in [12,13 ] 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.

Moreover, 3-D model is proposed to improve re-ID performance. A new 3-D model model called SARC3D [16] 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.

As for the 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 $8 * 8 * 8 = 512$ 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[14] 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 the scale-extrema of the function difference of Gaussian function with scale $\sigma$, that is,

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

the Gaussian filter with standard deviation $k1 * \sigma$, $I(x, y)$ is the image, when the DoG image is computed, each point $(x, y)$ is compared with its neighbours to get extrema.

MSCR (maximally stable color region) is also used in [6]. The MSCR is derived from MSER (maximally stable extreme region), it detects the region with stable color and uses an agglomerative clustering algorithm to compute color clusters, 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.

RHSP (Recurrent Highly-Structured Patches) is used in [6]. 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.

Combined descriptors are found to have better performance. Descriptors combining color and texture are most often used in re-identification. In [5] 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 [6]. Similar

to SDALF descriptor [6], 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

The second step of re-ID is to design the similarity computing methodology to compare descriptors. That is, the way to compare how different two descriptors are. This is also call the metric learning. Generally, for two input vectors $x_1$, $x_2$, any symmetric positive semi-definite matrix W defines a pseudo-metric with the form of $D = \boldsymbol{x}_1 * W * \boldsymbol{x}_2$. Many widely used distance metric obey this rule. Previous methods includes the Euclidean distance, Bhattacharyya distance and Mahalanobis distance. The Euclidean distance, which is mostly used for the straightforward descriptors like color and texture descriptors, is one case of the $L_p$ distance when $p = 2$, and also a special case of Mahalanobis distance when the covariance matrix of two input observations is an identity matrix. One example of metric learning is the probabilistic relative distance comparison model proposed in [4]. This model decreases the error caused by sometimes high intra-class variation and low inter-class variation. Compared with other distance learning models proposed, this model behaves more robust for its probability relative distance comparison model. 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 $f(.,.)$ so that $f(z, z') < f(z, z'')$. The contribution of this paper is that the author transfers the distance learning problem into 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

$$P(f(z, z') < f(z, z'')) = (1 + e^{(f(z-z')-f(z-z''))})^{-1} \tag{2.1}$$

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

$$f = \arg\min_{f} r(f, O) r(f, O) = -log(\Pi_{O_i} P(f(z - z') - f(z - z'))) \tag{2.2}$$

$O = \{O_i = (x_i^p - x_i^n)\}$ , $x_i^p$,$x_i^n$ 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-definite matrix, in this way the distance function learning problem is transformed to a matrix learning problem. The author used an iteration algorithm to compute matrix $M$.

Since still image based person representation suffers from factors like illumination, occlusion, viewpoint change and pose difference. The multi-shot re-ID has been proposed. Since there are a sequence of images for each indivudual, there are much more cues to exploit. In [TDL], 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 intra-class distance should be smaller than the minimal distance of inter-class distance. One another requirement is the sum of all intra-class distance should be as small as possible, so the final target function is summarized as

$$\begin{aligned} f(D) = (1 - \alpha) \sum_{x_i, x_j, y_i = y_j} D(x_i, x_j) + \\ \alpha \sum_{x_i, x_j, y_i = y_j} \max\{D(x_i, x_j) - \min_{y_i \neq y_k} D(x_i, x_k) + \rho, 0\} \end{aligned} \tag{2.3}$$

Cross view quadratic discriminant analysis(XQDA) is proposed in [xqda paper]. In [36 of lomo paper] Mogaddam et al. proposed to model each of two classes with a multivariate Gaussian distribution. Suppose the sample difference $\Delta = x_i - x_j$, where $x_i$ and $x_j$ are two feature vectors. $\Delta$ is called intrapersonal difference when their label $y_i = y_j$ and extrapersonal difference when $y_i \neq y_j$. Respectively two the intrapersonal and interpersonal variation can be defined as $\Omega_I$ and $\Omega_E$,

Besides those aforementioned metric learning, some metric learning by neural network also draws much interest. That is, to define a neural network to compute similarity of two input descriptors or even images. Recently the deep neural network has been exploited to improve the performance. One advantage of neural network

re-ID is the preprocessing of images can be skipped (We can also say the preprocess is included in convolutional layers). The input of this structure can be straightforward grey images or color images. To deal with multi-shots and video based re-identification neural network is proven to have better performance. But for the classical neural network there are too many weights to train and the over-fitting problem can be troublesome. Convolutional neural network can avoid those problems while remains 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 [11] for the first time 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 [17] 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 [18] 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. This method gives a solution to the problem that most CNNs are not trained enough caused from datasets with small number of images since this CNN learns from multiple datasets.

# Chapter 3

# Descriptors extraction

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 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 []. In this comparison camera a views are used as probe set and camera B views are used for gallery set. We can find that those color spaces separating intensity information outperform RGB color space and HSV outperforms all other color spaces.

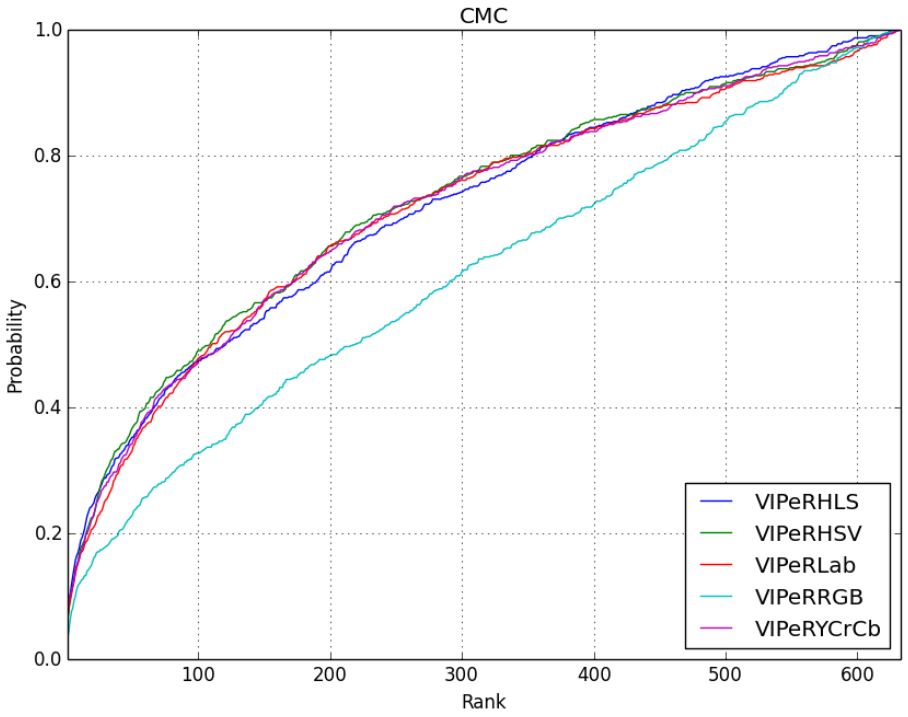


FIGURE 3.2: A CMC comparison of color histogram on different color spaces

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

[].

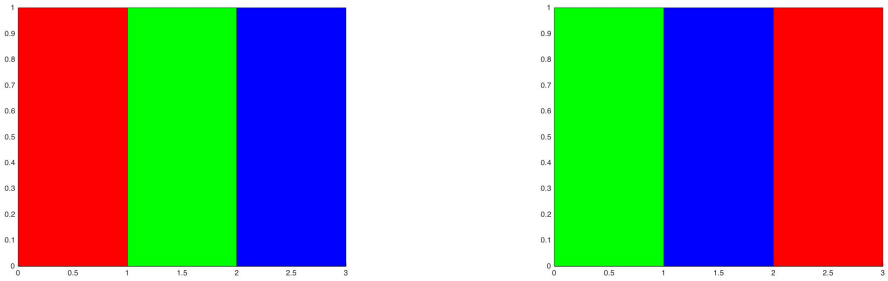

FIGURE 3.3: A comparison of two patches with same entropy but different color distribution

### 3.1.2 Local binary pattern(LBP)

Local binary pattern[] extracts the texture information with efficient computing and has been used on people detection and recognitions. Figure [] 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.

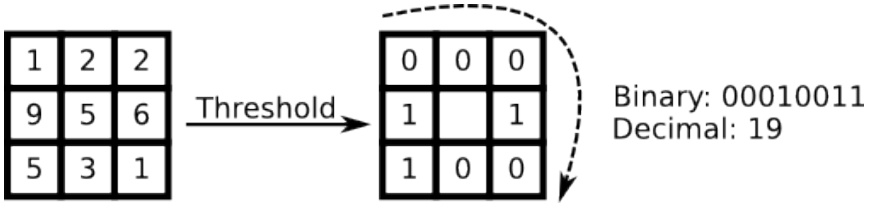


FIGURE 3.4: An LBP example, by thresholding the neighbour pixels the pixels are transformed into a binary number

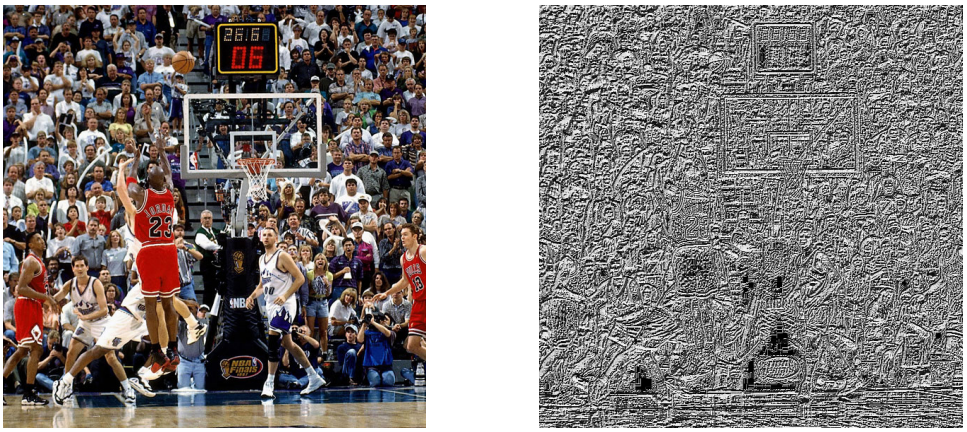

FIGURE 3.5: One LBP example

### 3.1.3 Histogram of oriented gradients(HOG)

The HOG [] descriptor also extracts textural information of images by gradient computing. Like many other descriptors, it computes the gradient of input images $I(x, y)$ first by equations

$$I_x = \frac{\partial I}{\partial x}, I_y = \frac{\partial I}{\partial y}, \tag{3.1}$$

the gradient can be computed fast by some discrete derivative maskd 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:

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

Then

## 3.2 Influence of background segmentation on different 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 [SDALF] the author provides foreground mask for all images. 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 [] and figure [].

FIGURE 3.6: Foreground segmentation of individuals from VIPeR

We can find that foreground segmentation decreases LBP's performance onVIPer dataset but increases HSV color histogram 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 handles texture for each sample patch, its performance suffers from those body parts loss and those little black patches.

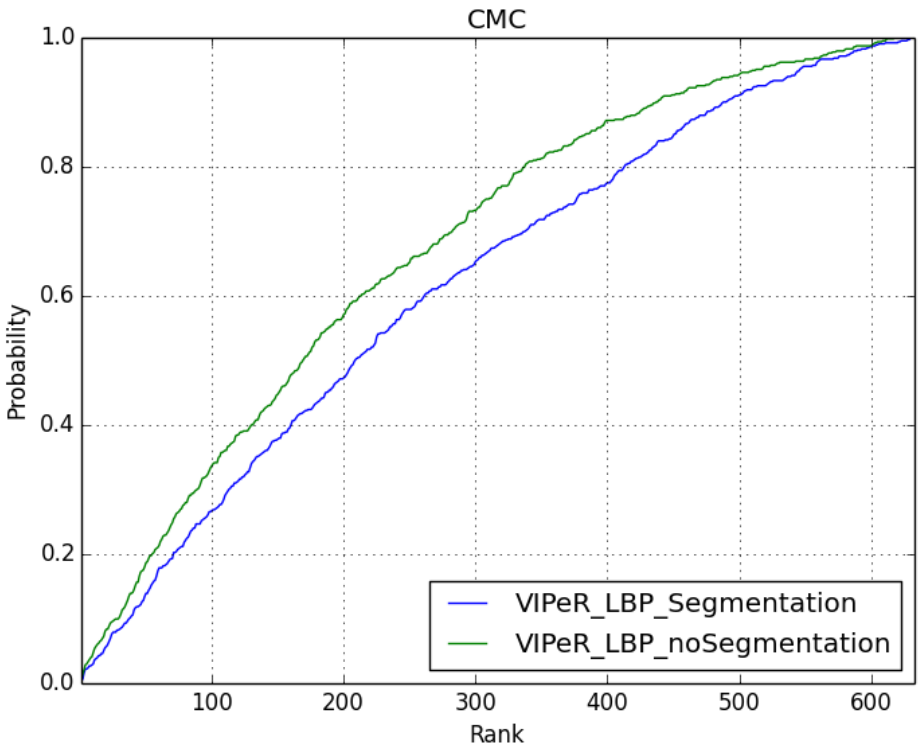


FIGURE 3.7: A CMC comparison of foreground segmentation on LBP tested on VIPeR

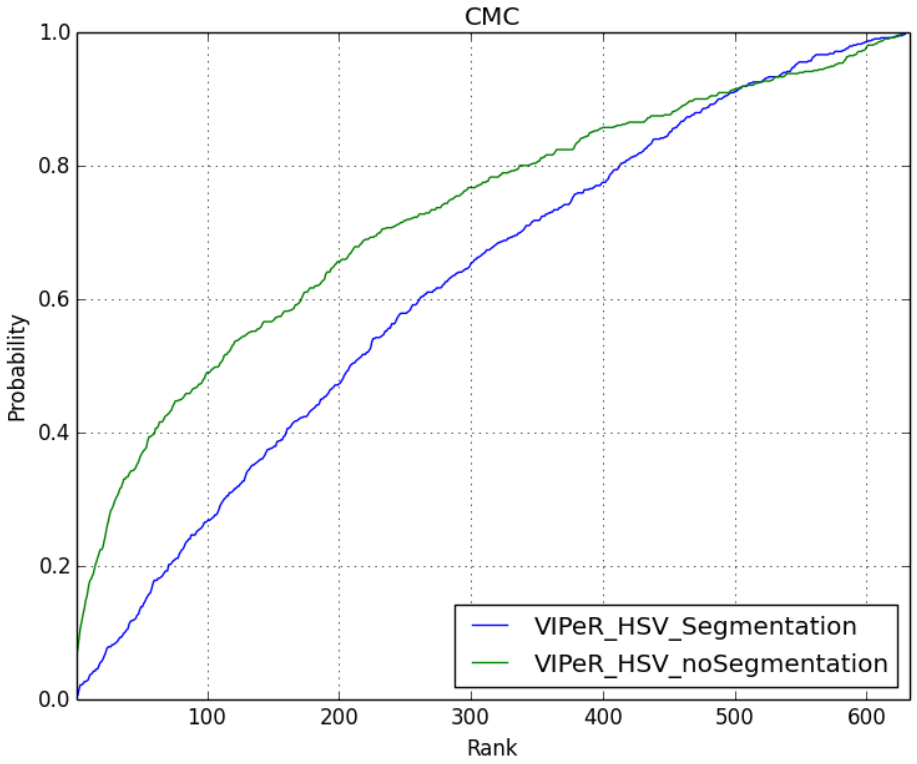


FIGURE 3.8: A CMC comparison of foreground segmentation on HSV histogram descriptor tested on VIPeR

## 3.3 The hierarchical gaussian descriptor

The hierarchical gaussian descriptor is proposed by in **GOGpaper** 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 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 **f**, 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. So 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

$$\begin{aligned} h_x &= [-1, 0, 1] \\ h_y &= -h_x' \end{aligned} \tag{3.4}$$

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:

$$\begin{aligned} O(i,j) &= (\arctan(I_y(i,j)/I_x(i,j)) + \pi) * 180/\pi \\ M(i,j) &= \sqrt{(I_x(i,j)^2 + I_y(i,j))^2} \end{aligned} \tag{3.5}$$

The orientation are quantized into four bins by a soft voting algorithm[ GOG15]. 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,

$$M_\theta = MO_\theta, \tag{3.6}$$

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

$$G(\boldsymbol{f}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{\exp^{(\frac{1}{2}(\boldsymbol{f}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{f}-\boldsymbol{\mu}))}}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|} \tag{3.7}$$

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

$$\boldsymbol{\mu} = \frac{1}{n}\sum \boldsymbol{f}_i, \tag{3.8}$$

$$\boldsymbol{\Sigma} = \frac{1}{n-1}(\boldsymbol{f}_i - \boldsymbol{\mu})(\boldsymbol{f}_i - \boldsymbol{\mu})^T, \tag{3.9}$$

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 $\mu$ and $\Sigma$.

### 3.3.2 Integral image for fast computation

To compute the estimated covariance matrix $\Sigma$, the integral image is used to reduce time complexity. The integral image[ ] 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

$$S(x', y') = \sum_{x<x', y<y'} I(x, y), \tag{3.10}$$

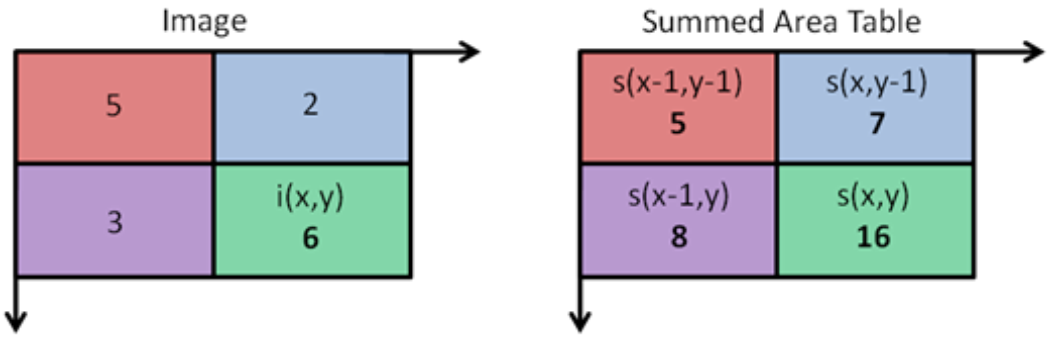


FIGURE 3.9: Integral image

### 3.3.3 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 previous level 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 [GOG25], the mapping can be denoted as

$$G(\boldsymbol{x}_i; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \sim \boldsymbol{P}_i = |\boldsymbol{\Sigma}_i|^{1/(d+1)} \begin{bmatrix} \boldsymbol{\Sigma}_i + \boldsymbol{\mu}_i \boldsymbol{\mu}^T & \boldsymbol{\mu}_i \\ \boldsymbol{\mu}_i^T & 1 \end{bmatrix} \tag{3.11}$$

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

After this mapping, the $n+1$ dimensional SPD matrix needs to be transformed as 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?and it can be compressed by half that only the upper triangular elements are preserved. To ensure the sum of 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

$$\boldsymbol{p}_i = [\boldsymbol{Q}_{1,1}, \sqrt{2}\boldsymbol{Q}_{1,2}, \sqrt{2}\boldsymbol{Q}_{1,3}, \cdots, \sqrt{2}\boldsymbol{Q}_{1,d+1}, \tag{3.12}$$

$$\boldsymbol{Q}_{2,2}, \sqrt{2}\boldsymbol{Q}_{2,3,}, \cdots, \sqrt{2}\boldsymbol{Q}_{2,d+1,}, \cdots, \boldsymbol{Q}_{d+1,d+1,}] \tag{3.13}$$

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 shown in [ ] 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}is extracted. The nRGB color space is calculated as

$$\begin{aligned} nR &= \frac{R}{R+G+B}, \\ nG &= \frac{G}{R+G+B}, \\ nR &= \frac{B}{R+G+B}, \end{aligned} \tag{3.14}$$

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

# Metric learning on subspace

## 4.1 Mahalanobis distance

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

$$D(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T \boldsymbol{M} (\boldsymbol{x} - \boldsymbol{y}), \tag{4.1}$$

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

$$D(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T \boldsymbol{W}^T \boldsymbol{W} (\boldsymbol{x} - \boldsymbol{y}) = ||\boldsymbol{W}(\boldsymbol{x} - \boldsymbol{y})|| \tag{4.2}$$

Therefore, Mahanalobis distance can be regarded as a variant of Euclidean distance. There are many methods proposed for metric learning[ ].

## 4.2 Gradient descent optimization

Given a multivariate function $F(\boldsymbol{x})$, if $f((x))$ is continuous and differentiable in the neighbour of point $\boldsymbol{x}$ for all $\boldsymbol{x}$, then $f((x))$ 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 use by updating $F$ with respect to $\boldsymbol{x}$. If the updating step $\lambda$ is small enough, by updating $\boldsymbol{x}$ with

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \lambda \boldsymbol{G} \tag{4.3}$$

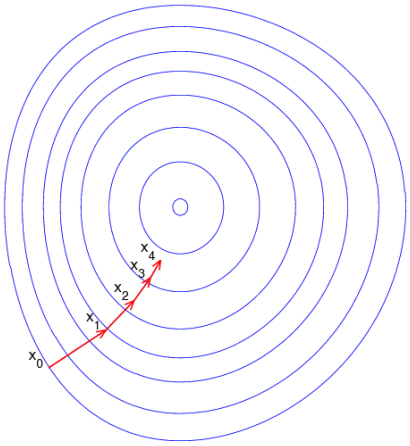


FIGURE 4.1: Steepest gradient descent

we have

$$F(\boldsymbol{x}_{t+1}) \geq F(\boldsymbol{x}_t). \tag{4.4}$$

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

**Analysis of steepest gradient descent method** The advantages of gradient descent are the 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. In this case the convergence value will depend on the initial value of $\boldsymbol{x}$. 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 []. The third shortcoming is linear search in gradient descent might cause some problem[].

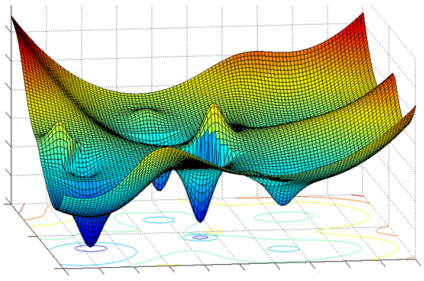

FIGURE 4.2: Function with multi local minimums

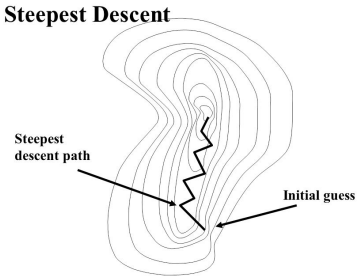


FIGURE 4.3: Zigzagging downhill valley

## 4.3 Metric learning based on sample pairs distance comparison

Inspired by You et al., 2015, 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 [PRDC], this method is also based on similarity comparison. The difference is in Zheng, Gong, and Xiang, 2016, 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 possible positive and negative pairs, computation complexity will be quite huge. To decrease complexity, a simplified version is proposed as the top-push distance metric learning[ ]. 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 smaller than the minimum of all negative distance. This can be denoted as

$$D(\boldsymbol{x}_i, \boldsymbol{x}_j) + \rho < \min D(\boldsymbol{x}_i, \boldsymbol{x}_k), y_i = y_j, y_i \neq y_k. \tag{4.5}$$

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

$$\min \sum_{y_i = y_j} \max\{D(\boldsymbol{x}_i, \boldsymbol{x}_j) - \min_{y_i \neq y_k} D(\boldsymbol{x}_i, \boldsymbol{x}_k) + \rho\}. \tag{4.6}$$

However, the equation above only penalize the interclass distance. Another term is needed to penalize intra class distance. That is, to make the sum of intraclass distance as small as possible. This term is denoted as

$$\min \sum D(\boldsymbol{x}_i, \boldsymbol{x}_j), y_i = y_j. \tag{4.7}$$

To combine equations above, a ratio factor $\alpha$ is assigned to equation [] so that the target function can be denote as

$$\begin{aligned} f(\boldsymbol{M}) = (1-\alpha) \sum_{\boldsymbol{x}_i, x_j, \boldsymbol{y}_i = y_j} D(\boldsymbol{x}_i, \boldsymbol{x}_j) + \\ \alpha \sum_{\boldsymbol{x}_i, \boldsymbol{x}_j, y_i = y_j} \max\{D(\boldsymbol{x}_i, \boldsymbol{x}_j) - \min_{y_i \neq y_k} D(\boldsymbol{x}_i, \boldsymbol{x}_k) + \rho, 0\} \end{aligned} \tag{4.8}$$

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

$$D(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T \boldsymbol{M} (\boldsymbol{x} - \boldsymbol{y}) = trace(\boldsymbol{M} \boldsymbol{X}_{i,j}) \tag{4.9}$$

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,

$$\begin{aligned} f(\boldsymbol{M}) = (1-\alpha) \sum_{y_i = y_j} trace(\boldsymbol{M} \boldsymbol{X}_{i,j}) \\ + \alpha \sum_{y_i = y_j, y_i \neq y_k} \max\{trace(\boldsymbol{M} \boldsymbol{X}_{i,j}) - trace(\boldsymbol{M} \boldsymbol{X}_{i,k}) + \rho, 0\} \end{aligned} \tag{4.10}$$

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

$$\boldsymbol{G} = \frac{\partial f}{\partial \boldsymbol{M}} = (1-\alpha) \sum_{y_i = y_j} \boldsymbol{X}_{i,j} + \alpha \sum_{y_i = y_j, y_i \neq y_k} (\boldsymbol{X}_{i,j} - \boldsymbol{X}_{i,k}) \tag{4.11}$$

The iteration process can be summarized as in table []

**Gradient optimization algorithm for target function**

**Input** Descriptors of training person pairs

**Output** A SPD matrix

**Initialization**

Initialize $\boldsymbol{M}$ 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 $t = t + 1$;

Update gradient $\boldsymbol{G}_{t+1}$ with equation 24;

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-definite space

by $\boldsymbol{M}_{t+1} = \boldsymbol{V}_{t+1}\boldsymbol{S}_{t+1}\boldsymbol{V}_{t+1}^T$;

Update the target value $f|_{\boldsymbol{M}=\boldsymbol{M}_{t+1}}$;

**end while**

return $\boldsymbol{M}$

# 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 $48 \times 128$. 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 $60 \times 160$. In this paper, we randomly select 485 image pairs as training data and the rest person pairs are used for test data.

**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.

**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.

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 [GOG], 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 [2,3,4,5,6], 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 | 43.32 | 74.78 | 85.00 | 89.94 | 93.39 |
| Preprocessed GOGrgb | 43.73 | 74.75 | 85.41 | 90.28 | 93.86 |
| Original GOGfusion | 48.67 | 77.41 | 87.41 | 91.65 | 94.34 |
| Preprocessed GOGfusion | 48.10 | 76.90 | 87.59 | 91.90 | 94.40 |

TABLE 5.3: The influence of data preprocessing on CUHK1

| | Rank(%) | | | | |
|---|---|---|---|---|---|
| Terms | 1 | 5 | 10 | 15 | 20 |
| Original GOGrgb | 56.15 | 83.79 | 90.08 | 92.63 | 94.26 |
| Preprocessed GOGrgb | 55.86 | 84.28 | 90.45 | 93.09 | 94.65 |
| Original GOGfusion | 57.00 | 84.55 | 90.37 | 92.82 | 94.69 |
| Preprocessed GOGfusion | 56.69 | 84.40 | 90.53 | 93.27 | 94.90 |

TABLE 5.4: The influence of data preprocessing on prid_2011

| | Rank(%) | | | | |
|---|---|---|---|---|---|
| Terms | 1 | 5 | 10 | 15 | 20 |
| Original GOGrgb | 24.70 | 51.80 | 63.30 | 69.60 | 72.70 |
| Preprocessed GOGrgb | 23.80 | 52.10 | 63.50 | 70.20 | 73.50 |
| Original GOGfusion | 32.40 | 56.80 | 66.80 | 73.10 | 77.70 |
| Preprocessed GOGfusion | 32.20 | 57.50 | 66.40 | 73.50 | 78.00 |

TABLE 5.5: The influence of data preprocessing on prid_450s

| | Rank(%) | | | | |
|---|---|---|---|---|---|
| Terms | 1 | 5 | 10 | 15 | 20 |
| Original GOGrgb | 61.02 | 84.22 | 91.33 | 94.09 | 96.22 |
| Preprocessed GOGrgb | 60.44 | 84.44 | 91.33 | 94.00 | 96.13 |
| Original GOGfusion | 62.89 | 86.62 | 92.53 | 95.29 | 96.89 |
| Preprocessed GOGfusion | 62.62 | 86.44 | 92.36 | 95.20 | 96.93 |

TABLE 5.6: The influence of data preprocessing on GRID

| | Rank(%) | | | | |
|---|---|---|---|---|---|
| Terms | 1 | 5 | 10 | 15 | 20 |
| Original GOGrgb | 22.96 | 42.00 | 51.76 | 58.72 | 64.64 |
| Preprocessed GOGrgb | 22.80 | 43.76 | 52.08 | 59.04 | 65.12 |
| Original GOGfusion | 24.32 | 44.40 | 54.96 | 62.40 | 66.56 |
| Preprocessed GOGfusion | 23.84 | 44.64 | 55.04 | 62.24 | 66.24 |

## 5.3 Parameters setting of gradient descent iteration

In this experiment, there are a few parameters for the iteration computing including slack variable $\rho$, maximal iteration $T$, gradient step $\lambda$, the inter and intra class limitation factor $\alpha$ and the updating ratio $\beta$. Firstly the slack variable $\rho$ is initialized as 1 to ensure the minimum inter class distance is 1 larger than intra class distance at least. The step size of gradient updating $\lambda$ is initialized as 0.01. When target value $f$ increases, $\lambda$ is scaled by a factor 0.5, and $\lambda$ is scaled by 1.01 when target value

$f$ decreases. To judge if target value converges, the thresh $\beta$ 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 $\alpha$, to know the best value for $\alpha$, we tried 11 different values ranges from 0 to 1 with a step of 0.1, and the rank-1 and rank-5 scores of responding $\alpha$ is shown in figure []. The best $\alpha$ value should have as large top rank scores as possible. By comparison, $\alpha$ is set as 0.7. A form of all parameters are shown in Form 7.

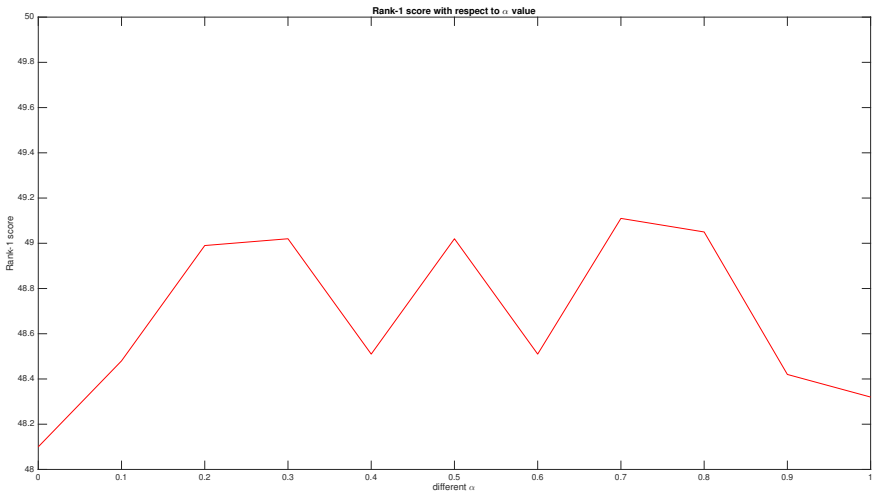


FIGURE 5.1: Rank 1 scores with respect to $\alpha$ on VIPeR

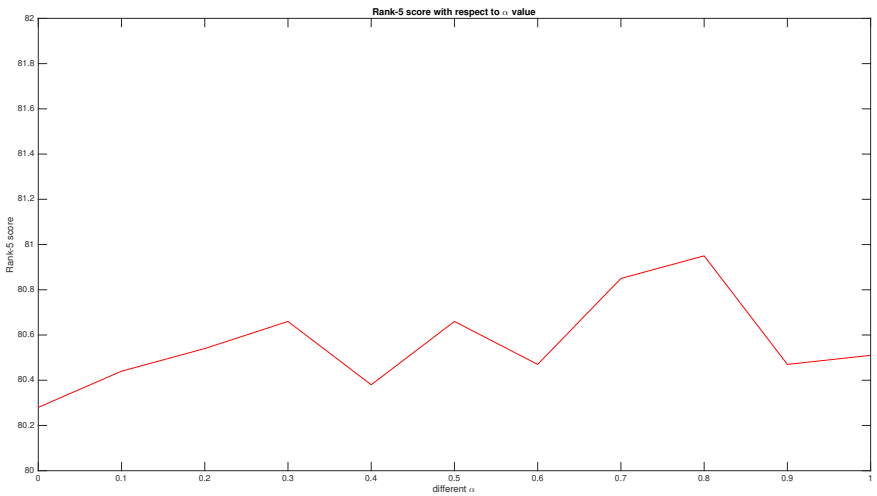


FIGURE 5.2: Rank 5 scores with respect to $\alpha$ on VIPeR

TABLE 5.7: Parameters setting

| Paramters | $\alpha$ | 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 perfect CMC curve is expected to have a high rank-1 value and reaches 1 as fast as possible.

## 5.4 Performance analysis

In this paper, we compare proposed metric with other state-of-the-art metrics including NFST[], XQDA[]. 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 $M$ 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 8. 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 GOGrgb scores of proposed metric learning are 0.44%, 0.72%, 1.27%, 1.3%, 1.47% 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 0.19%, -0.79%, 0.86%, 0.63%, 0.67% higher than GOGfusion + XQDA respectively. Also we can see that the proposed metric learning has a way more better performance than NFST. We can infer that the performance of KLFDA is better than XQDA, with its rank-1 score 1.6% higher.

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 | 43.48 | 74.59 | 85.35 | 90.47 | 93.67 |
| 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 | 48.16 | 76.65 | 87.66 | 91.90 | 94.37 |

**CUHK1** We can find that the rank 1, rank5, rank 10, rank 15, rank 20 score of GOGrgb combined with proposed metric are 5.35%, 4.22%,3.35%,2.1%,1.44% higher than XQDA, and 0.26%,1.26%,1.38%,1.11%, 1.09% than NFST. Also the rank 1, rank5, rank 10, rank 15, rank 20 score of GOGfusion combined with proposed metric are 4.59%, 2.55%, 0.72%, 1.29%, 0.89% higher than GOGfusion combined with XQDA, and 0.4%, 0.74%, 0.6%, 1.05%, 1.2% than GOGfusion combined with NFST.

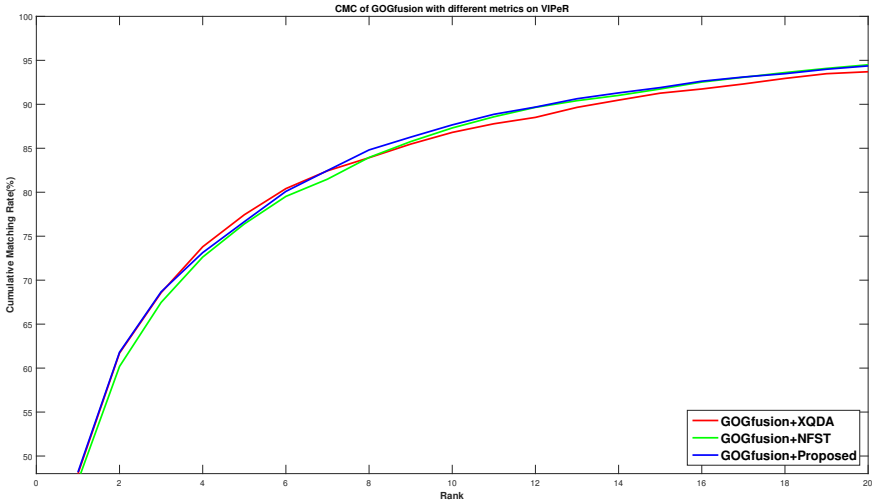


FIGURE 5.3: CMC curves on VIPeR comparing different metric learning

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 | 55.86 | 84.28 | 90.45 | 93.09 | 94.65 |
| 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 | 56.69 | 84.40 | 90.53 | 93.27 | 94.90 |

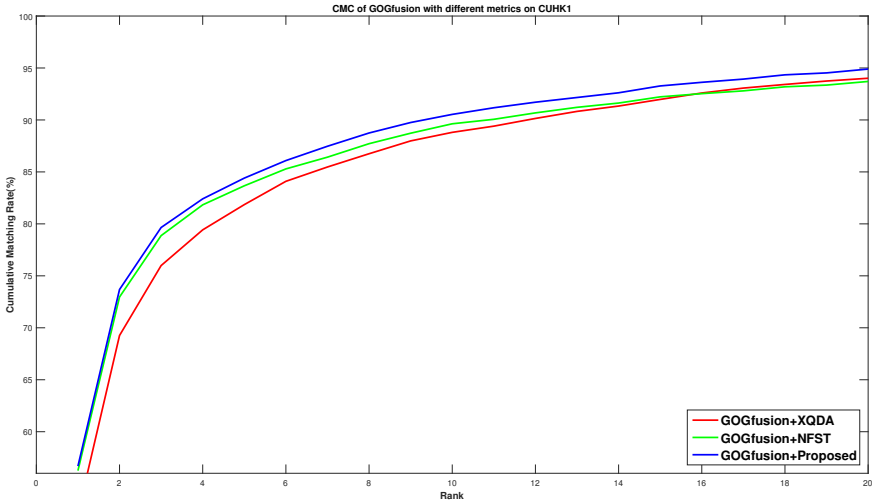


FIGURE 5.4: CMC curves on CUHK1 comparing different metric learning

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 | 52.10 | 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 | 32.20 | 57.50 | 66.40 | 73.50 | 78.00 |

**Prid_2011** The rank 1, rank5, rank 10, rank 15, rank 20 score of GOGfusion combined with proposed metric are 6.2%, 3.8%, 4.4%, 2.1% 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.

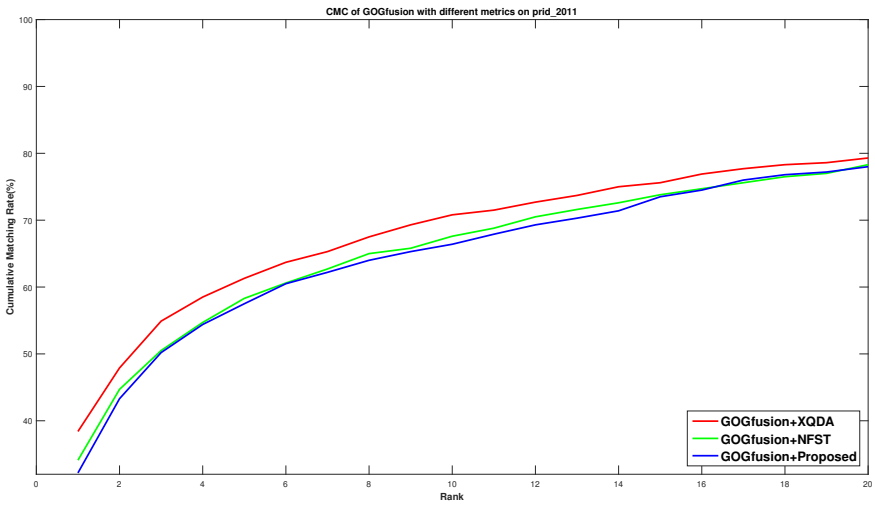


FIGURE 5.5: CMC curves on prid_2011 comparing different metric learning

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 | 60.44 | 84.44 | 91.33 | 94.00 | 96.13 |
| 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 | 62.62 | 86.44 | 92.36 | 95.20 | 96.93 |

**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 descriptors.

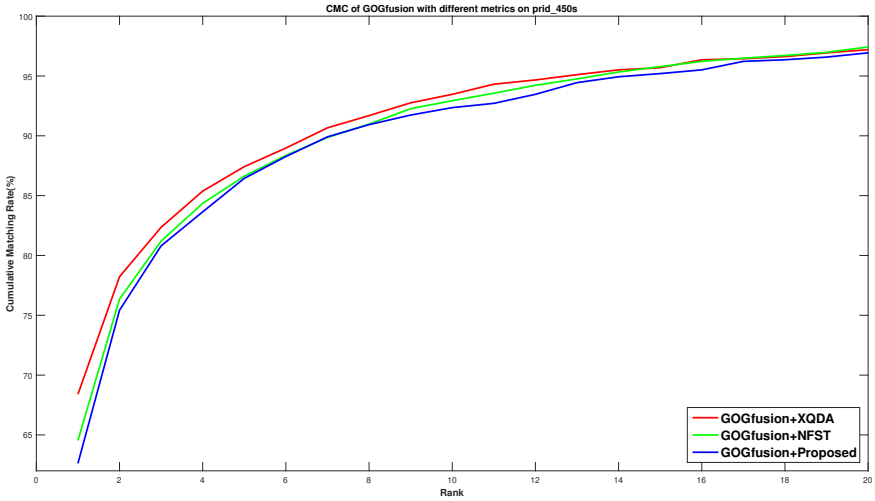


FIGURE 5.6: CMC curves on prid_450s comparing different metric learning

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 | 22.80 | 43.76 | 52.08 | 59.04 | 65.12 |
| 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 | 23.84 | 44.64 | 55.04 | 62.24 | 66.24 |

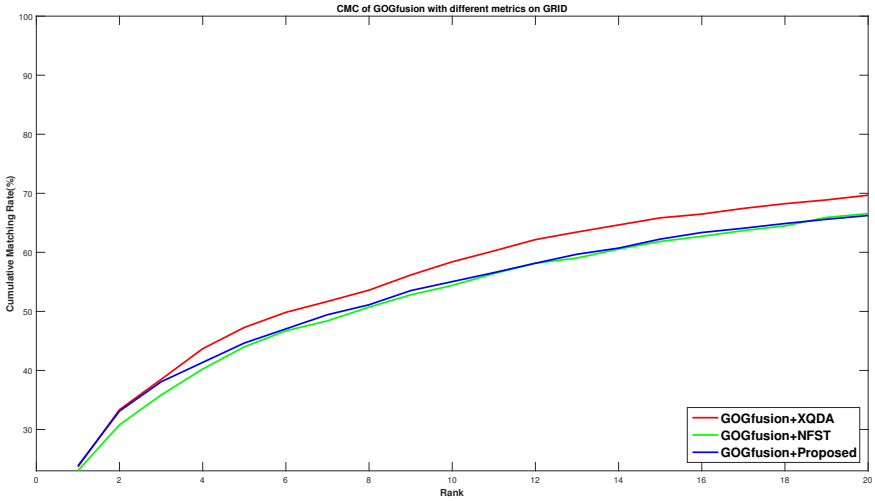


FIGURE 5.7: CMC curves on GRID comparing different metric learning

**GRID** We can see that the rank 1 score of proposed metric are slightly higher than XQDA and 0.8% higher than NFST in terms of GOGfusion, but XQDA outperforms proposed metric on rank 5, rank 10, rank 15 and rank 20 scores. But proposed metric outperforms NFST on rank 5, rank 10, rank 15 and rank 20 scores.

# 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

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

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