PERSON RE-IDENTIFICATION WITH PAIRWISE LEARNING AND RANKING

A Thesis presented to

the Faculty of the Graduate School

at the University of Missouri

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

---

by

ZHI ZHANG

Dr. Zhihai He, Thesis Supervisor

MAY 2014

The undersigned, appointed by the Dean of the Graduate School, have examined the dissertation entitled:

PERSON RE-IDENTIFICATION
WITH PAIRWISE LEARNING AND RANKING

presented by Zhi Zhang,
a candidate for the degree of Master of Science and hereby certify that, in their opinion, it is worthy of acceptance.

Dr. Zhihai He, Ph.D, Department of Electrical and Computer Engineering

---

Dr. Tony Xu Han, Ph.D, Department of Electrical and Computer Engineering

---

Dr. Ye Duan, Ph.D, Department of Computer Science

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my research advisor Prof. Zhihai He for the continuous support of my study and research, with his patience, enthusiasm, immense knowledge and ideas. Without his guide throughout the process, I would never have the chance to finish this thesis.

Besides my advisor, I would also like to show my deepest thankfulness to my committee: Dr. Tony Xu Han and Dr. Ye Duan for their encouragement and insightful comments, especially the invaluable knowledge in Computer Vision during my first year of attendance in University of Missouri. Their teaching style and enthusiasm made a strong impression on me and lead me into this field.

Special thanks goes to my fellow lab-mates, also good friends, in Communication and Video Processing lab: York Chung, Xiaobo Ren, Chen Huang, Yifeng Zeng, Guanghan Ning, Di Wu, Yizhe Zhu and Jingxin Ou, for the discussion we had, for the genius suggestions they provided, and for the enjoyable time we spent together. I would also like extend my thanks to Dr. Wenpeng Ding, Dr. James Keller, Dr. Marjorie Skubic, Dr. Alina Zare, Dr. Michela Becchi, Dr. Guilherme DeSouza and Dr. Jeffrey Uhlmann for their kind support related to my study and research.

Last but not the least, I am grateful to my family, especially to my parents, who give birth to me and support me throughout my life.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

ABSTRACT

In this work, we address the problem of person re-identification for intelligent analysis and search of surveillance videos. During person re-identification, we need to match observations of individuals across different camera views with large variations of appearance, such as pose, illumination, and view angle. We develop a system that combines traditional color histograms and proposed semantic color descriptors with our local features encoded by Fisher vectors, to provide discriminative appearance-based representation of persons. We also develop an adaptive training sample selection schemes to optimize the training performance over a large scale training dataset. A two-layer linear ranking SVM with pruning method is introduced to handle such a large training set. At last, the result is represented by a ranking score over all gallery images given a probe image. We evaluated our system performance on VIPeR dataset and compared to previous results, demonstrating the effectiveness and the robustness of our methods against significant environmental changes.

Chapter 1

Introduction and Background

1.1 Introduction

Person Re-identification, which is also known as inter-camera human association or multi-camera tracking, as one of the most important applications in computer vision, has been an active research field for decades [54]. Recently, especially during the past few years, person re-identification system has received more attention as a result of its increasing importance in a wide range of practical applications in both commercial and law enforcement areas [57]. Human Re-identification techniques have been or potentially will be adopted by many applications such as surveillance, identification systems, video retrieval and search, *etc* [27].

Person Re-identification mainly involves the following two steps: descriptor extraction based on appearance and distance matching between the candidates. In the first step, two assumptions are applied [1]: 1) The fine cues (*e.g.* face, or iris, which

are commonly used for identification problems) are not available due to the relatively low resolution of the captured images; 2) The individuals across different cameras do not change their clothes. For the distance matching step, given an query image and a set of candidates, the target image (*i.e.* query person in a different scene) should have the closet distance in terms of descriptors when paired with query image. The pairwise distance metric can be achieved by unsupervised or supervised learning(if ground-truth provided). In this work, we will concentrate on the latter.

Though many re-identification systems have demonstrated promising results under well-controlled settings [32, 58], such as small scale CCTV networks, person re-identification itself remains a complicated problem that is far from being completely solved in wild conditions. There are two major difficulties that make human re-identification in uncontrolled environments a very challenging problem. The first is due to the relatively large intra-personal variation under different combinations of backgrounds, human poses, illumination and view angles. All these facts are quite obvious in real world scenario. Moreover, different cameras with uncalibrated sensor parameters will introduce unpredictable errors. Second, distinct people may present similar appearances as long as they wear clothes with the same color. Figure 1.1 shows some examples of the images captured by camera A and B in VIPeR dataset [2].

1.2 Related Works

In order to tackle this problem, many existing works concentrate on two major approaches: descriptor/feature extraction and distance measurements.

Figure 1.1: VIPeR dataset sample images. Top row from cam_a, bottom row from cam_b. Each column: same subject from different viewpoints.

1.2.1 Feature Representation

In the first approach, the visual features applied in person re-identification are complicated, which involve tons of works and analysis, however, they can still be roughly categorized into global and local descriptors. The holistic representation of body parts and detailed traits are both significant, plus there are so many existing and new methods to handle with, thus no doubt the way to select and combine various global and local features plays the most important role in this part. Some typical and widely adopted features for this problem include colors [3, 4, 1], Histogram of Oriented Gradients (HoG) [5, 6, 7], Haar-like descriptors [8], points of interest [9], *e.g.* SIFT and SURF [10, 11], Maximally Stable Color Regions (MSCR) [12, 13, 14], texture filters [2, 15, 16, 17], differential local information [2], co-occurrence matrices [6]. Gray *et al.* [2] introduced their feature combination including RGB, YCbCr, HSV, Schmid and Gabor filters. They also evaluated the percent of descriptors their

model selected among all those features, providing interesting results that every channel is supporting the final feature space, while hue and saturation channels are most informative without surprise given the illumination changes between two cameras. As we can conclude, at least two to three lower-level features are combined in each of these works. For example, Prosser *et al.* [17] merged colors and textures and Kuo [1] concatenated multi-channel color histograms with MSCR and Covariance Matrices.

Despite the low level features that are directly extracted from the images, some approaches like Bag of Words (BoW) [18] are commonly used to further utilize the statistical informations. In BoW, the visual word occurrences are recorded and used to present the target image. The histogram features based on trained visual words code-book are very robust to spatial changes, thus be widely adopted in computer vision problems. The BoW model has been introduced for person re-identification in [10], where contextual information enriched visual words are embedded and grouped as descriptors. In recent years, the introduction of the Fisher vector [19] provides a better model to encode the local features. Many works [20, 21, 22] have shown the outstanding performance of Fisher vectors over other coding mechanisms.

1.2.2 Distance Measure

Typical general distance measures include histogram based Euclidean distance, Bhattacharyya distance [17], K-Nearest Neighbor classifiers [23]. To be more discrimination oriented, Gray and Tao [2] proposed to use Adaptive boosting algorithm (Adaboost) [24] to search for the most relevant features through the entire set of descriptors based on the assumption that certain features often appear to be more suitable for matching then others. Re-identifications by these approaches both encounter dif-

ficulties. The rationales are complex. For the former one, note that not every piece of descriptors are equally important,and there are always highly overlapped feature distributions of different objects. For example, in Figure 3.3, given a probe image, an incorrect gallery image can appear to be more similar to the probe than a correct image in the gallery. The challenge is hard to be solved by any of these low-level distance matching methods. In contrast, the adaptive boosted weak classifiers keep a small portion of features that are considered the most discriminative. However, this method become less effective when similar negative samples have severely overlapped feature distributions, which leads to a confusion of the weak classifiers and the ignorance of informative descriptors.

Reformulation to ranking problem of re-identification have been introduced by many works. In RankBoost [25], Freund *et al.* uses a set of weak rankers boosted to form a strong ranker. This method was also adopted by Kuo [1]. Unlike RankBoost, Joachims *et al.* seek to learn a ranking function based on SVM kernels in a much higher dimensional space where features are more separable. However, the main issue with the kernel based RankSVM learning is the comparative expensive computation due to the huge amount of inequality constraints introduced by its super high dimensionality. As a consequence, it is limited to a few iterations, resulting in sub-optimal solution. A primal-based RankSVM (PRSVM) [26] was propsed by Chapelle and Keerthi to address the speed issue of RankSVM. However, the PRSVM still suffers another scalability limitation problem. More specifically, as the number of training samples grows, the number of negative samples increases non-linearly.[1]

Distance learning and matching methods are receiving increasing concentration.

[1]Negative samples grows exponentially in most cases according to positive samples.

Pairwise metric learning (RPLM) [27] is proposed based on Mahalanobis distance [28] which takes advantages of the structure of the data with reduced computational cost. Kostinger *et al.* [29] proposed a simple method to learn the distance metric based on a statistical inference perspective. Zheng *et al.* formulate re-identification problem as a relative distance comparison (PDRC) problem which aims to maximize the likelihood that the distance between a pair of images of the same pedestrian is smaller than a incorrect pair.

To summarize, there is no optimal distance measurement learning solution for every possible application, in other words, all learning methods are task oriented. Either they assume the training data to be fully annotated, or they are too domain specific [30, 31, 32], or they suffer from significant loss in performance when the dimensionality of the input space is high or amount of available training data is low [33, 34].

1.3 Overview of This Work

In this work, our framework is organized into two fold: *training* and *testing*. They share the same feature extraction module to ensure the consistency of descriptors extracted from every image. Besides, *training* is served as a prerequisite for *testing* since distance learning weight vector is required for pairwise ranking. Figure 1.2 illustrate the flowchart of our proposed method.

We aim to seek for the most discriminative features while preserve the global information collaboratively; Also, to provide a similarity score rather than correct *vs.* incorrect classification and utilize the massive pairwise information. Thus, in

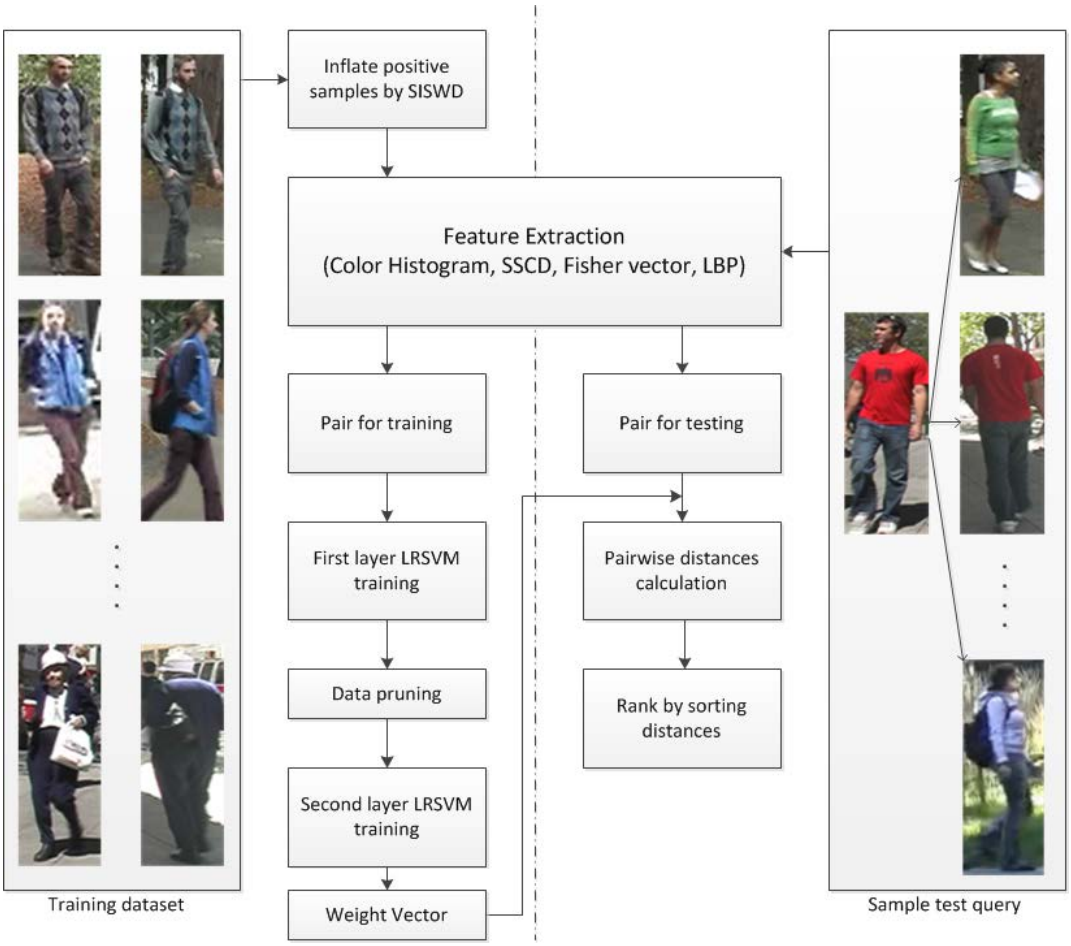

Figure 1.2: The overview flowchart of our work.

Section 2.1, we discuss the color histograms and introduce our semantic color name descriptor as well as its statistical extension SSCD. A higher level encoding method Fisher vector is adopted in Section 2.2, followed by the proposed texture descriptor uniform Local Binary Patterns and our combination mechanism in Section 2.3 and 2.4 respectively. We also propose our distance learning method including Self Inflation by Shifting Window Duplicates (SISWD), Linear Ranking Support Vector Machine (LRSVM) and Pruning for data discrimination refine in Chapter 3.

To evaluate the effectiveness of our partial and compete method, we conducted

experiments on the feature channels and compared our performance with some state-of-art results in Chapter 4. It demonstrates that our framework in pedestrian re-identification is effective and could achieve state-of-art performance.

Chapter 2

Viewpoint Invariant Image Representation

Extracting the representation of image appearance is fundamental yet important part of most computer vision problems. To begin with extraction, note that a pedestrian image is bounded in a rectangle. Corresponding to human body shape, the height is larger than the width of a given image. Each image will be divided into 6 non-overlapping strips, all the feature extraction steps presented below are performed on each strip. The justification is provided in Section 2.4.

2.1 Color Descriptors

As discussed before, the significance of color appearance in this problem is no doubt obvious. To fully utilize this information, we propose three separate formulations: color histograms, semantic color names and statistical descriptor based on color

names.

2.1.1 Color Histograms

A color histogram is a representation of the distribution of colors on a specific color channel. The original RGB channels are prone to illumination changes, as a result, we include HSV and Lab channels to maintain the smooth color transitions in complex lighting conditions and disjoint camera views. Without loss of generality, we assume in a given color space c containing K color bins, the color histogram of image strip I containing N pixels is represented as $H(I) = [h_1, h_2, ..., h_n]$, where $h_i = n_i/N$ is the number of a pixel in the image strip belonging to the i-th color bin, and n_i is the total number of pixels in the i-th color bin. h_i can be defined as follows:

$$h_i = \sum_{j=1}^{N} P_{i|j} \tag{2.1}$$

where $P_{i|j}$ is the quantization and selection function:

$$P_{i|j} = \begin{cases} 1 & \text{if } i = 0 \ \& \ I_{min} \leq I_i \leq Wi \\ 1 & \text{if } 0 < i \leq K \ \& \ W(i-1) < I_i \leq Wi \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

where $W = I_{max}/K$ is the span of color bins.

To compute the histograms $H(I)$ in a given image strip, we first convert each pixel from RGB to HSV and Lab, then quantized to 16 bins separately in each channel, result in $16 \times 3 \times 3 = 144$ dimensional features.

2.1.2 Semantic Color Descriptors

Instead of using histograms only as in many previous works, we added semantic color names feature to address the issue that inter-camera pixel values of the same person may have small offsets. We have noticed that when comparing a pair of images, people use color names to classify a certain range of colors and ignore the slight difference in values. For instance, two pixels A (255,0,0) and B (220,0,0) are both 'red' in terms of human visual acknowledgment. However, their RGB values have a large gap. The semantic color names are considered to be more robust and less sensitive to noise. To obtain the names, we need a mapping function from the HSV color space values to color names in our dictionary. We choose HSV instead of RGB values because HSV channels are much more likely to stay contiguous when lighting condition changes. Figure 2.1 shows the well defined bright color distribution in HSV color-space based on hue angles, which are listed in Table 2.1.

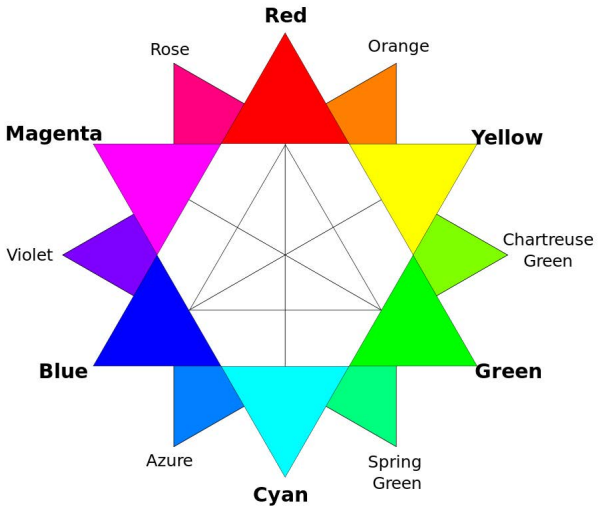

Figure 2.1: The semantically labeled bright color wheel in HSV space based on hue angles.

Angle	Color
0°	Red
30°	Orange
60°	Yellow
90°	Chartreuse green
120°	Green
150°	Sprint green
180°	Cyan
210°	Azure
240°	Blue
270°	Violet
300°	Magenta
330°	Rose

Table 2.1: The hue angles of the 12 major bright colors of the HSV color wheel.

We use 15 color names in English [35, 36]: black, gray, white, red, orange, yellow, chartreuse green, green, sprint green, cyan, azure, blue, violet, magenta, rose. The first three are pale colors, while the rest are vivid and can well represent the bright color of the clothes.

Kuo *et al.* [1] use a training method assuming the prior probabilities are equal among all color names to get a mapping function from every RGB value to 11 color names. We did not follow their methodology and the rationale is quite simple: color names are human defined semantic feature, we need the labels based on ground-truth.

We first normalize hue, saturation and value to range $[0, 1]$, and the pseudo-code for this is shown in Algorithm 1. Figure 2.2 shows the output of semantic color names. Note that the colors in these images are painted only for viewing, the actual name labels are exclusive to any value. The semantic color names labeling is more stable than the original color values even though there are some artifacts due to the complex mixture of cloth textures and shadow.

Algorithm 1 Semantic Color Name

Require: $h, s, v \in [0, 1]$
  if $v < 0.3$ **then**
    $name \leftarrow BLACK$
  else
    if $s < 0.2$ **then**
      if $v \in [0.3, 0.8)$ **then**
        $name \leftarrow GRAY$
      end if
      if $v \geq 0.8$ **then**
        $name \leftarrow WHITE$
      end if
    else
      $SWITCH(h)$
      $name \leftarrow BRIGHT\_COLOR\_NAME\_BASED\_ON\_HUE\_ANGLE$
    end if
  end if
  return name

2.1.3 Statistical Semantic Color Descriptor

To suppress the error introduced by large pose variation, we also include pose invariant features. Statistical Semantic Color Descriptor (SSCD) is thresholded percentage based on semantic color names descriptor. For each color name we introduced in the last section, such as blue, we will have M binary bins. The SSCD for this specific color is a simple thresholding function:

$$SSCD_{blue}(m) = \begin{cases} 1 & \text{if } sum_{blue}/sum_{total} \geq \frac{m}{M} \\ 0 & \text{otherwise} \end{cases}, m = 0, 1, 2, ...M. \tag{2.3}$$

where sum_{blue} is the summation of pixel numbers that were defined as "blue", and sum_{total} is the summation of pixels in current strip. M is the quantization parameter which also controls the number of bins.

Figure 2.2 shows the results of images before and after SSCD, some shadows are removed. We can observe that the bulk flat color areas become even more smooth.

Figure 2.2: Some examples of pixel-wise assignments by SSCD on VIPeR dataset. (a) Original samples. (b) Result images. Note that the painted colors are used only for demonstration, the descriptors are index numbers only in practice.

In practice, we set $M = 10$, which results in $15 \times 10 = 150$ bins SSCD in each strip, where 15 is the number of semantic color names.

2.2 Fisher Vectors

To further exploit the local low-level details and their combination patterns, we use Fisher vectors to encode the properties of images. First, we need a very simple descriptor, then a generative model such as Gaussian Mixture Model (GMM), and finally an encoder that will generate a frequency based representation whose idea is similar to the BoW model. Overall, the Fisher vector characterizes a signal with a

gradient vector derived from a probability density function (PDF), models not only the generative but also discriminative patterns at the same time.

2.2.1 Local Descriptor

We use intensity, first and second order derivatives to represent the local traits. For each channel in HSV space, the local descriptor consists of 5 elements:

$$f(x,y) = (I(x,y), I_x(x,y), I_y(x,y), I_{xx}(x,y), I_{yy}(x,y)) \tag{2.4}$$

where x and y are coordinates, $I(x,y)$ is the intensity at (x,y) of corresponding channel, I_x and I_y are the first-order derivatives with respect to x and y, while I_{xx} and I_{yy} are the second-order derivatives. Given a color image consists of three channels, this result in $5 \times 3 = 15$ dimensional point-wise descriptor. We can see this local descriptor captures the absolute value as well as the gradient change, contains the majority of information restricted within a local region.

2.2.2 Gaussian Mixture Models

A Gaussian Mixture Model is introduced to approximate the distribution of low-level features in images, *i.e.* a visual word or vocabulary. Let us denote $D = \{d_n, n = 1, 2, ...N\}$ be the set of the N local descriptors extracted from images. We model D with a Gaussian Mixture Model using Maximum Likelihood(ML) estimation. Let $\hat{\mu}_\lambda$ be the GMM model:

$$\hat{\mu}_\lambda = \sum_{k=1}^{K} w_k \mu_k(\mu_k, \sigma_k) \tag{2.5}$$

where K is a controllable parameter of the number of Gaussian components, w_k denotes the weight of the $k-th$ Gaussian component, while μ_k and σ_k are its mean and standard deviations. The Gaussian Mixture Models can be obtained by maximizing the log-likelihood of the extracted local features:

$$\ell(\Theta; X) = E_{x\sim\hat{p}}[\log p(x|\Theta)] = \frac{1}{n}\sum_{i=1}^{n}\log\sum_{k=1}^{K} w_k p(X_i|\mu_k, \Sigma_k) \tag{2.6}$$

where $\hat{p}$ is the empirical distribution of the data. We can use Expectation Maximization (EM) [37] to solve (2.6) since the direct maximization of the log-likelihood function of a GMM is difficult due to the fact that the assignments of points to Gaussian mode is not observable and, as such, must be treated as a latent variable. By introducing an auxiliary distribution $q(h|x)$ on the latent variable, we can use Jensen's inequality to obtain the lower bound on the log-likelihood:

$$\begin{aligned}
\ell(\Theta; X) = E_{x\sim\hat{p}}\log p(x|\Theta) &= E_{x\sim\hat{p}}\log\int p(x,h|\Theta)dh \\
&= E_{x\sim\hat{p}}\log\int \frac{p(x,h|\Theta)}{q(h|x)}q(h|x)dh \\
&\geq E_{x\sim\hat{p}}\int q(h)\log\frac{p(x,h|\Theta)}{q(h|x)}dh \\
= E_{(x,q)\sim q(h|x)\hat{p}(x)}\log p(x,h|\Theta) &- E_{(x,q)\sim q(h|x)\hat{p}(x)}\log q(h|x)
\end{aligned} \tag{2.7}$$

where the first term of the last expression is the log-likelihood of the model, the second term is the average entropy of the latent variable, which does not depend on Θ. Then iterative expectation step:

$$q_{ik} = \frac{w_k p(x_i|\mu_k, \Sigma_k)}{\sum_{l=1}^{K} w_l p(X_i|\mu_l, \Sigma_l)} \tag{2.8}$$

and maximization step:

$$\mu_k = \frac{\sum_{i=1}^{n} q_{ik} X_i}{\sum_{i=1}^{n} q_{ik}} \tag{2.9}$$

$$\Sigma_k = \frac{\sum_{i=1}^{n} q_{ik} (X_i - \mu_k)(X_i - \mu_k)^T}{\sum_{i=1}^{n} q_{ik}} \tag{2.10}$$

$$w_k = \frac{\sum_{i=1}^{n} q_{ik}}{\sum_{i=1}^{n} \sum_{l=1}^{K} q_{il}} \tag{2.11}$$

are performed to estimate the GMM.

2.2.3 Fisher Vector Encoding

After the GMMs are obtained, we can encode the image using Fisher vector. Chatfield *et al.* [22] have demonstrated that Fisher vector outperforms the BoW model by a large margin, indicating the fact that information is lost when a local descriptor is replaced with (assigned to) the nearest codeword. In contrary, Fisher vector successfully retains extra information by soft assignment to the Gaussian components. Let $\gamma_t(k)$ be the soft assignment of the descriptor d_t to the Gaussian component k:

$$\gamma_t(k) = \frac{w_k \mu_k(d_t)}{\sum_{j=1}^{K} w_j \mu_j(d_t)} \tag{2.12}$$

$G^M_{\mu,k}$ and $G^M_{\sigma,k}$ are the 15-dimensional descriptor gradients with respect to μ_k and σ_k of the component k. $G^M_{\mu,k}$ and $G^M_{\sigma,k}$ can be computed via the following derivations:

$$G^M_{\mu,k} = \frac{1}{T\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) (\frac{m_t - \mu_k}{\sigma_k}) \tag{2.13}$$

$$G_{\sigma,k}^{M} = \frac{1}{T\sqrt{2w_k}} \sum_{t=1}^{T} \gamma_t(k)[\frac{(m_t - \mu_k)^2}{\sigma_k^2} - 1] \tag{2.14}$$

where the divisions between vectors are term-by-term operations.

We concatenate $G_{\mu,k}^{M}$ and $G_{\sigma,k}^{M}$ vectors for $k = 1, 2, ...K$, resulting in a $2 \times 15 \times K$ dimensional feature, where 15 is the dimensionality of our local descriptor.

2.3 Local Binary Patterns

So far, we have included local and statistical color features, as well as color gradients information. In terms of gray-scale texture descriptors, we choose Local Binary Patterns (LBP) [38] because it is proved to be very effective in capturing local patterns such as edges, spots and flat areas.

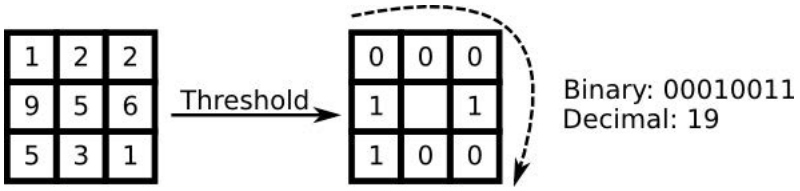

Figure 2.3: A typical Local Binary Pattern operator.

Regardless of scale invariant setting, a typical LBP value is calculated within a 3 x 3 block. In Figure 2.3, a center pixel value 5 from the left box is compared to its 8 neighbors to generate binary codes denoted as F in the right box:

$$F \approx t(s(g_0 - g_c), s(g_1 - g_c), ..., s(g_{P-1} - g_c)) \tag{2.15}$$

where $P = 8$ in this case and

$$s(x) = \begin{cases} 1 & x \geq T \\ 0 & x < T \end{cases} \tag{2.16}$$

where T is the threshold parameter. If not specifically required, we normally set $T = 0$.

The next step is to concatenate the binary codes in clockwise order, the highest bit starting from top or top-left corner. Note that points $P = 8$ and radius $R = 1$ for the rest parts if not specified and as long as the order is fixed and consistent, the starting position does not affect the performance:

$$LBP_{P,R}(x, y) = \sum_{p=0}^{P-1} s(g_p - g_c)2^P \tag{2.17}$$

For example, in Figure 2.3, for the center pixel in the 3 x 3 block, its LBP value $LBP_8(2, 2) = 0 * 2^7 + 0 * 2^6 + 0 * 2^5 + 1 * 2^4 + 0 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0 = 19$. The basic LBP operator could generate $8 - bit = 256$ unique identifiers.

The basic LBP is very sensitive to a rotation or a tilt of the source image because in which case, the relative positions have changed between g_c and g_p. To remove the potential effect of rotation, *i.e.*, to assign a unique pattern to each rotational invariant LBP [39], we can define:

$$LBP^{ri}_{P,R} = min[ROR(LBP_{P,R}, i)|i = 0, 1, ...P - 1] \tag{2.18}$$

where function $ROR(x, i)$ performs a circular bit-wise right shift on the $P-bit$ number x for i times. In terms of the neighboring pixels, (2.18) simply do the iteratively

clockwise rotation until a maximal number of the most significant bits[1], starting from g_{P-1}, is 0.

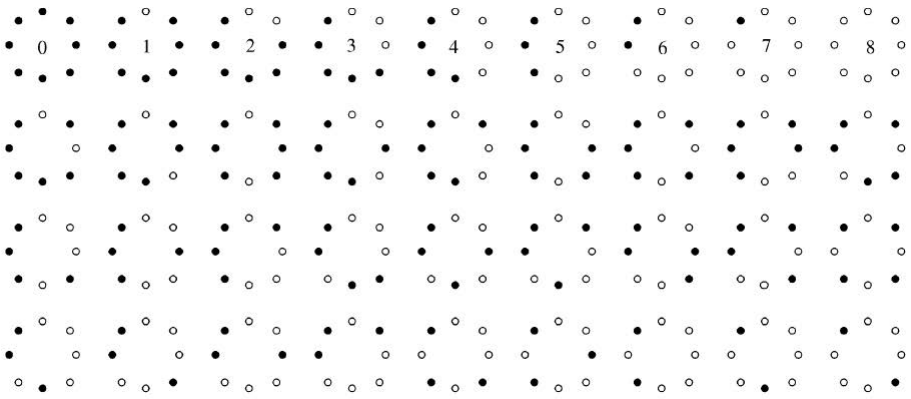

Figure 2.4: The 36 unique rotation invariant local binary patterns. Black and white circles correspond to 0 and 1 in the 8-bit operator output respectively.

Observations [38] have shown the basic and rotation invariant version do not provide good discrimination with two rationales: The occurrence frequencies of the 36 unique patterns vary greatly and the rough quantization of the angular intervals of 45°. Experimental results indicate a certain set of local binary patterns are fundamental elements of texture, account for the vast majority, sometimes more than 90 percent of the 3 x 3 patterns in the observed textures set. Mäenpää *et al.* [40] call the fundamental set "uniform" patterns as they have one thing in common, namely, uniform circular structure that contain very few spatial transitions. Uniform patterns can effectively present bright spot(0), flat area or dark spot (8), edge or corner (1-7) as illustrated on the first row of Figure 2.4.

To formally define whether a specific pattern x is uniform, a uniformity measure, $U(x)$ in (2.20) is defined as the number of spatial transitions in x. Only patterns with

[1]Here significant means larger weight 2^P, not the importance of this bit.

equal to or less than two bitwise transitions are uniform patterns. Thus, the uniform LBP operator, LBP_P^{u2} is defined as:

$$LBP_{P,R}^{u2} = \begin{cases} I(LBP_{P,R}(x,y)) & \text{if } U(LBP_{P,R}) \leq 2 \\ (P-1)P+2 & x < T \end{cases} \tag{2.19}$$

where

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)| \tag{2.20}$$

and $I(z) \in [0, (P-1)P+2)$, $(P-1)P+2$ is the total number of uniform patterns that satisfy the constraint. The superscript $u2$ in (2.19) indicates that the $U(x)$ is restricted to be smaller than 2. And the index function $I(z)$ is used to assign a particular index to each of the uniform patterns. For those patterns whose U is larger than 2, they are all mapped to the same index $(P-1)P+2$ which is a trivial and complementary bin.

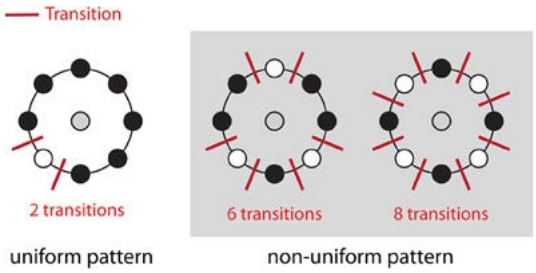

Figure 2.5: The illustration of binary transitions in LBP, indicated by intersect lines.

For instance, patterns 00000000_2 and 11111111_2 have U equals to 0, while in Figure 2.5, the left pattern 00000100_2 has $U = 2$ because it has transition $(0 \to 1)$ and $(1 \to 0)$ each for once, and the middle and right patterns have U values of 6 and 8

respectively. In this case, the left LBP is considered a uniform pattern while the rest two are not. The non-uniform patterns will be mapped to the same bin 58, where 58 correspond to the last index of $(8-1)\times 8+2$ patterns in terms of eight neighborhoods LBP operator. We categorize and plot the 58 patterns in 2.6 motivated by [41].

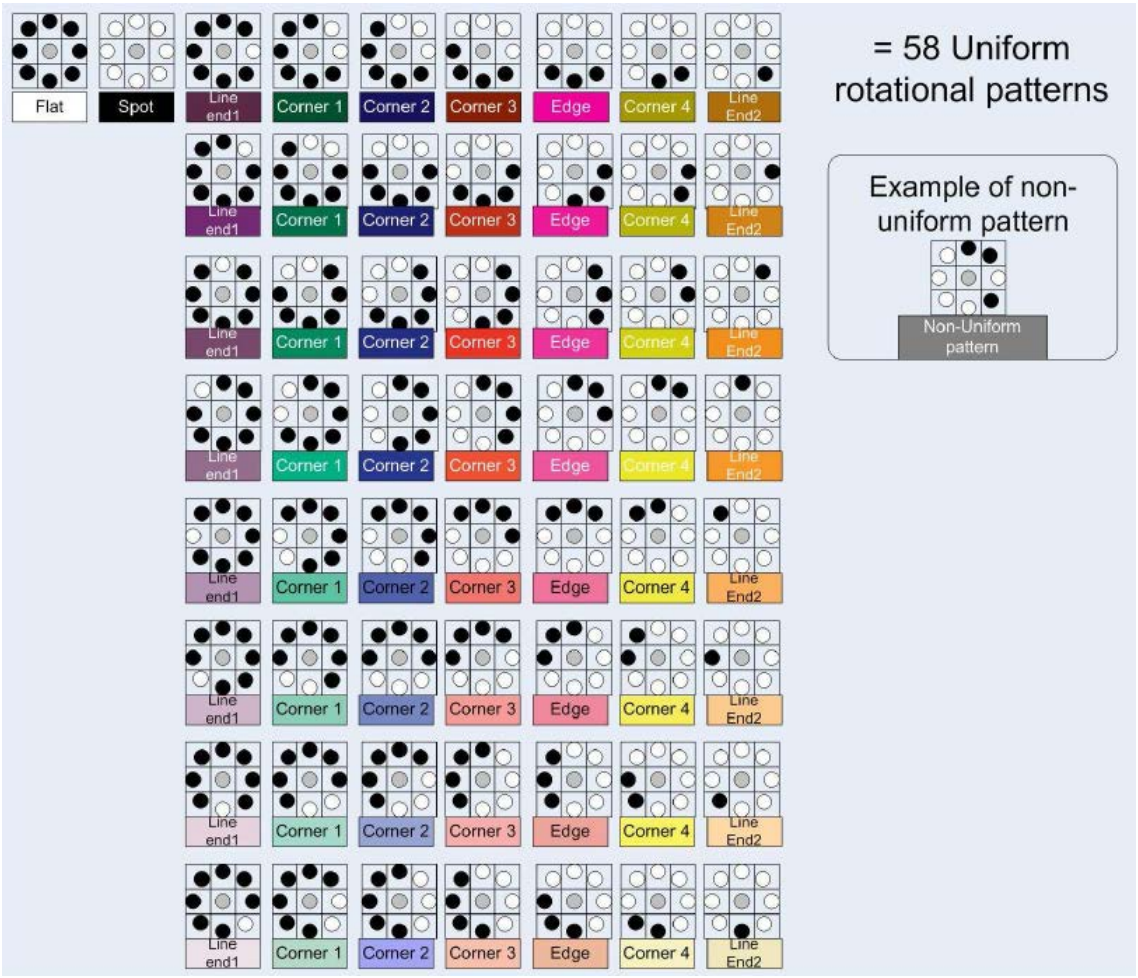

Figure 2.6: The categorized 58 uniform local binary patterns.

In Figure 2.6, each uniform pattern has different properties, such as flat, spot, corner and line tip. Thus we can explain LBP in a more intuitive way, that is, local texture patterns.

The local binary patterns must be voted to bins (histogram) to take effect, since strict pixel-wise assignment of LBP does not make any sense here in this problem because a tiny movement of the pedestrian in an image will change the LBP value greatly. Only statistical histogram can correctly reflect the real appearance in a specific region, or a strip, as will be discussed later.

Technically, uniform LBP operators are very computational efficient to achieve by using a 256-D look-up table.

2.4 Ensemble of Features

As we have mentioned, an image was divided into 6 non-overlapping strips. The motivation is the uncertainty of horizontal axis result from the flexible view angles, while the vertical structure of body, for example, the relationship between head and torso, keeps intact. Figure 2.7 shows a pair of images from the same person. We can see dramatically changed horizontal contents. By cutting along X coordinate, we preserve the vertical information while eliminate the horizontal variant introduced by pose and view angle shift.

In each strip, we concatenate the all the existing descriptors into one feature. More specifically, for color histograms, RGB, HSV each counts for 48 dimension and Lab color space takes the rest 41-D; For Fisher vector, we set $K = 30$ which is the number of Gaussian components, result in $30 \times 15 \times 2 = 900-$D FV, where 15 is the dimensionality of our local descriptor. The dimensionality of color name histogram seems to be relatively small, but it is the prerequisite of SSCD descriptor, and we will prove that it is very discriminative and effective despite its dimensionality.

Descriptor	Dimension
Color histograms	144
Color Name histogram	15
SSCD	150
Fisher vector	900
Uniform LBP histogram	58
Overall	1267

Table 2.2: This dimensionalities of different descriptors selected by us.

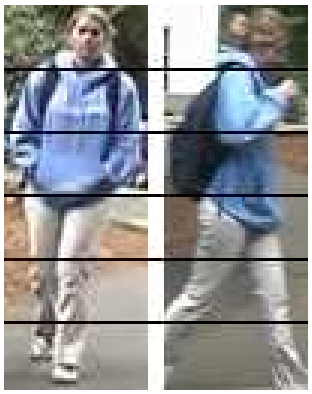

Figure 2.7: An example of 6 non-overlapping strips in a pair of images from the same person.

Due to the fact that LBP are not available at the borders and in order to utilize the benefit of background extraction in the future, we apply strip-wise L-1 normalizations. In this scheme, we can depress the effect of different sizes of extraction regions to the minimum. Finally, a $1267 \times 6 = 7602$ dimensional feature will be extracted from each query image and plugged into our training machine for further discrimination analysis.

An overview our feature extracted in each image is illustrated in Figure 2.8.

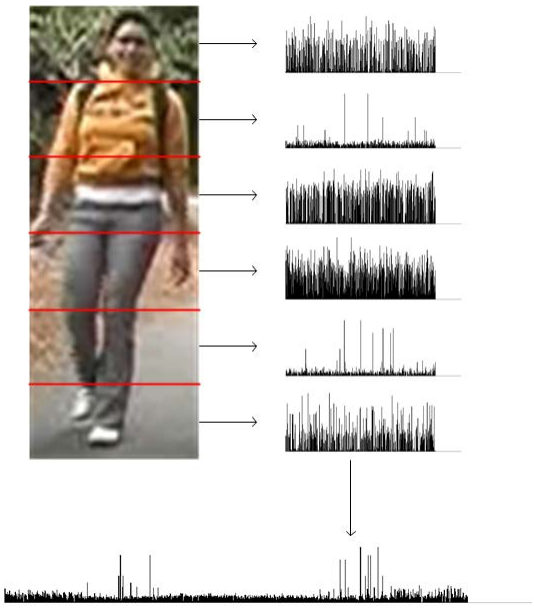

Figure 2.8: The overview of combined feature extraction on a given image.

Chapter 3

Pairwise Ranking

In the last chapter, we achieved discriminative image representations on every image in the training set. In this chapter, we will use the pairwise information to perform analysis and selection in the entire feature space.

3.1 Data Organizing

Since we have extracted features from every image in the training dataset, we need to pair each two images in order to generate a inquiry.

3.1.1 Feature Pairing

Let us denote $F_m(i)$ and $F_n(i)$ to be the i-th feature from $image_m$ and $image_n$ respectively. The pairwise feature $F_{m,n}(i)$ is given by:

$$F_{m,n}(i) = \frac{|F_m(i) - F_n(i)|}{|F_m(i)| + |F_n(i)| + 1} \tag{3.1}$$

where i is the feature space index. In the last chapter, we have mentioned that $i = 1, ..., N$, where $N = 7560$ is the dimensionality of our image descriptor. Thus, an inquiry pair is then given by:

$$P(m, n) = \{L(m, n), F_{m,n}(i)\}, i = 1, 2, ..., N \tag{3.2}$$

where $L(m, n)$ is the labeling function:

$$L(m, n) = \begin{cases} 1 & \text{if } image_m \text{ and } image_n \text{ indicate the same person} \\ -1 & \text{otherwise} \end{cases} \tag{3.3}$$

Let X be the number of individuals in a dataset. Assume a dataset contains 2 images for each individual, by computing (3.2) on every unique image pairs, we can get X intra-personal pairs, labeled by 1, whilst the number of inter-personal pairs is $(C_X^2 - X)$. If we take VIPeR dataset as an example, which has 632 pedestrian, the number of overall inter-personal image pairs is 198764; In contrast, we can only extract 632 intra-personal pairs. It is obvious that the data is severely imbalanced, which may lead to defect training result. This is also mentioned in [26].

3.1.2 Training Data Balancing

In order to balance the number of positive and negative samples, most works [42, 43, 1, 44] have chosen to use partial inter-personal samples and thus a large portion of information was wasted. Here we propose an approach called Self Inflation by Shifting Window Duplicates (SISWD) that is to inflate the intra-personal pairs rather than throwing out negative pairs. SISWD is based on the assumption that the general properties of an image will not change in case the image shifts with a small offset.

Figure 3.1: Illustration of SISWD process. The shifting window starting from red box, scanning with step size until orange box. The blue box is regarded as the original sample used for inter-personal pairs.

To use SISWD, a padding method or Region of Interest (ROI) should be applied to maintain the boundary of shifting windows. We use ROI and set the shift range from $\{-2, -1, 0, +1, +2\}$ for horizontal and $\{-1, 0, +1\}$ for vertical direction. See Figure 3.1 , a shifting window W starting from $(-2, -1)$ to $(2, 1)$ with step size 1 for both x and y directions, resulting $5 * 3 - 1 = 14$ duplicates and 1 original sample.

Therefore the number of intra-personal pairs within two images which belong to the same person is $C_1 5^2 = 105$. The illustration of SISWD is shown in Figure 3.1.

Note that the inflated duplicates are only used for intra pairs, the inter pairs are generated using the original ROI, *i.e.* shifting window $(0, 0)$, thus the number of inter-personal pairs will not be affected. Table 3.1 provides the comparison of number of samples before/after SISWD, as well as which from other works given 316 training images. Obviously, the imbalance problem is alleviated, however, the number of samples is even more large as the cost to contain as much pairwise information as we can.

Samples	Before	LDFV[42],LF[43]	Ours
Intra-personal	316	316	33180
Inter-personal	198764	632	198764
Total	199396	948	231944

Table 3.1: This Table shows the comparison of number of pairs used for training.

3.2 Linear SVM Ranking

Kernel based Ranking Support Vector Machine (KRSVM) [15] is a great tool for solving the re-identification ranking problem. However, with the large data challenge, some sacrifices such as lower precision, divided dataset and less iterations have to be made to achieve applicable results. We have studied a lot supervised learning techniques in recent years, according to Chapelle *et al.* [45] and others, the state of art learning methods that train ranking models can be categorized into three types. Point-wise methods, such as decision tree and linear regression, will directly learn the relevance score from each instance independently; Pairwise methods like RankSVM

[46] learn based on preference pairs; List-wise methods, for example, LambdaMART [47], are trying to optimize for the whole list. Of course, these types are not highly restricted, some methods lie between two categories, like GBRank [48], which combines point-wise decision tree models and pairwise loss. Motivated by Lee *et al.* [49], finally we focus on the Linear Ranking Support Vector Machine (LRSVM) which is efficient and thus suitable for training large-scale data.

3.2.1 Mathematical Formulation

Notation	Explanation
w	The weight vector obtained by solving (3.5)
x_i	The feature vector of the i-th training instance
y_i	Label for the i-th training instance
q_i	Query of the i-th training instance
K	The set of relevance levels
Q	The set of queries
P	The set of preference pairs; see (3.4)
l	Number of training instances
k	Number of relevance levels
p	Number of preference pairs
n	Number of features
$\bar{n}$	Average number of non-zero features per instance
l_q	Number of training instance in a given query q
k_q	Number of relevance levels in a given query q
T	An order-statistic tree

Table 3.2: Notation of LRSVM.

We list the notations in Table 3.2, assume we are given a set of training tuples (y_i, q_i, x_i), $y_i \in K = \{-1, +1\}$ is obtained by labeling function (3.3), $q_i \in Q$, $x_i \in R^n$, $i = 1, ..., l$, where Q is the set of queries. By defining the set of preference pairs as:

$$P \equiv \{(i, j) | q_i = q_j, y_i > y_j\} \; with \; p \equiv |P| \tag{3.4}$$

L2 loss function is used to optimize the problem by minimizing the objective training losses:

$$\min_{w} \quad \frac{1}{2} w^T w + C \sum_{(i,j)\in P} max(0, 1 - w^T(x_i - x_j))^2 \tag{3.5}$$

The summation of training losses can be written in the following separable form:

$$\sum_{q\in Q} \sum_{(i,j):q_i=q_j=q, y_i>y_j} max(0, 1 - w^T(x_i - x_j))^2 \tag{3.6}$$

As mentioned in the previous section, the large number of pairs in the loss term is the major difficulty to handle this problem. In this section, we consider the truncated Newton method to see what kind of information it requires.

A Newton method obtains a direction at the t-th iteration by solving

$$\min_{s} \quad g_t(s) \tag{3.7}$$

where

$$g_t(s) \equiv \nabla f(w^t)^T s + \frac{1}{2} s^T \nabla^2 f(w^t) s \tag{3.8}$$

and updates w^t by

$$w^{t+1} = w^t + s \tag{3.9}$$

Note that $g_t(s)$ is the second-order Taylor approximation of $f(w^t + s) - f(w^t)$. If $\nabla^2 f(w^t)$ is invertible, the step s is obtained by solving the following linear system:

$$\nabla^2 f(w^t) s = -\nabla f(w^t) \tag{3.10}$$

To ensure the convergence, usually line search or trust region techniques is applied

to obtain a truncated Newton step.

We consider a trust region Newton method (TRON) [50, 51, 52] that finds the direction s by minimizing $g_t(s)$ in (3.8) over a region we trust:

$$\min_s \quad g_t(s) \quad \text{subject to } ||s|| \leq \Delta_t \tag{3.11}$$

where Δ_t is the size of the trust region. After solving (3.11), it decides whether to apply the obtained direction s^t according to the approximate function reduction $g_t(s)$ and the real function decrease. Which means:

$$w^{t+1} = \begin{cases} w^t & \text{if } \rho_k < \eta_0 \\ w^t + s & \text{if } \rho_k \geq \eta_0 \end{cases} \tag{3.12}$$

where $\eta_0 \geq 0$ is a pre-specified parameter and

$$\rho_k \equiv \frac{f(w^t + s^t) - f(w^t)}{g_t(s^t)} \tag{3.13}$$

TRON adjusts the trust region Δ_t according to ρ_k. When ρ_k is too small, Δ_t will decrease, and vice versa. More specifically, the following rule is considered during TRON:

$$\Delta_{t+1} \in \begin{cases} [\sigma_1 \min(||s^t||, \Delta_t), \sigma_2 \Delta_t] & \text{if } \rho_k \leq \eta_1 \\ [\sigma_1 \Delta_t, \sigma_3 \Delta_t] & \text{if } \rho_k \in (\eta_1, \eta_2) \\ [\Delta_t, \sigma_3 \Delta_t] & \text{if } \rho_k \geq \eta_2 \end{cases} \tag{3.14}$$

where the parameters are provided in Table 3.3 according to [49].

Regarding the stopping criteria, we follow that of [53] to check if the gradient is

Parameter	Value
η_0	10^{-4}
η_1	0.25
η_2	0.75
σ_1	0.25
σ_2	0.5
σ_3	4.0

Table 3.3: The configuration of TRON parameters.

relatively smaller than the initial value:

$$||\nabla f(w^k)||_2 \leq \epsilon ||\nabla f(w^0)||_2 \tag{3.15}$$

where w^0 is the initial iterate and ϵ_s is the stopping tolerance that is a tunable parameter.

3.2.2 Distance Ranking

For a given pair of images, the pairwise distance is given by:

$$D_{m,n} = \sum_{i=1}^{N} F_{m,n}(i) w_i \tag{3.16}$$

where w_i is the i-th value of the weight vector obtained by solving (3.5). For a given testing query image m, the $rank\ r = 1$ image in the gallery set is defined as:

$$rank_1 = \arg\min_{i \in gallery} D_{m,i} \tag{3.17}$$

Similarly, we can obtain $rank\ r = n$ image given a probe image. CMCs are the statistics of the probability that given a probe image, the corresponding self image is

ranked in top n.

3.3 Pruning Training Data

LRSVM can outperform naive distance matching method by a large margin without doubt. However, to further eliminate the error introduced by bad samples, for example, dissimilar positive pairs (see Figure 3.2) and similar negative samples (see Figure 3.3, the image appearances in the middle are so close to the query image, sometimes even looks more similar than the target image itself), we propose a method to prune our existing training data.

Figure 3.2: The sample pairs of dissimilar images from the same person: Two images in each column correspond to the same individual. Top: images from camera a; Bottom: images from camera b.

For illustration, let us assume the distributions of real similarity (according to human) in training dataset are overlapping normal distributions, see Figure 3.4. Regions A-D have their unique effects on distance learning which will be reflected by final weights vector: For region A, negative pairs that come from different persons have smaller

similarity score, which is good and contribute to the correct negative samples. It is the set we want to keep; Region B is derived from those pairs we illustrated in Figure 3.2. These pairs are the target set we want to prune because they introduce the confusion to LRSVM that positive pairs are not necessarily to be visually similar. Similarly, region D corresponds to the informative positive samples with large similarity score, and region C denotes the bad samples to be removed.

Here we propose a two-layer LRSVM coupled with pruning to refine the training dataset and provide better result, which is illustrated in Figure 3.5.

The first layer LRSVM is used to train a initial weight vector w_0 as mentioned in the previous section. With w_0, we can compute the similarity score of every training sample using (3.16). Two parameters p^+ and p^- controls the percentage of positive and negative training samples to be kept after pruning respectively. To implement this, we sort positive and negative samples independently. Let us denote N_p to be the number of positive training samples, and N_n is the number of negative samples. Given the sorted result $sort_p(x)$ from large to small similarity score, and $sort_n(x)$ from small to large, we can define the selection function $F_p(x)$ for the positive samples:

$$F_p(x) = \begin{cases} 1 & \text{if } sort_p(x) < N_p * p^+ \\ 0 & \text{otherwise} \end{cases}, x \in Q_{positive} \tag{3.18}$$

and similarly, the selection function for the negative samples:

$$F_n(x) = \begin{cases} 1 & \text{if } sort_n(x) < N_n * p^- \\ 0 & \text{otherwise} \end{cases}, x \in Q_{negative} \tag{3.19}$$

The parameters p^+ and p^- can be achieved using 10-fold cross-validation. After

pruning, the new dataset will replace the original, and the second layer LRSVM is applied to obtain our final weight vector.

Figure 3.3: The sample pairs of similar images from different pedestrians. Left and right: query image and its target. Middle: similar images from different pedestrians.

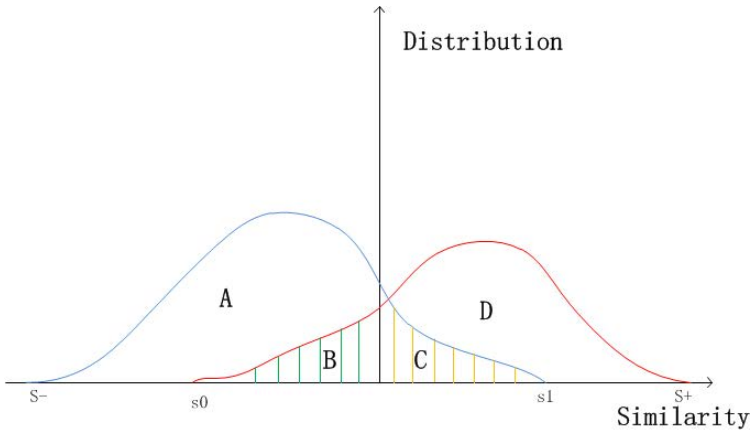

Figure 3.4: The similarity score distributions according to human definition. The blue curve to the left: similarity distribution of negative pairs; The red curve to the right: similarity distribution of positive pairs. For simple illustration, normal distribution is used, however, the real distributions are much more complex in practice.

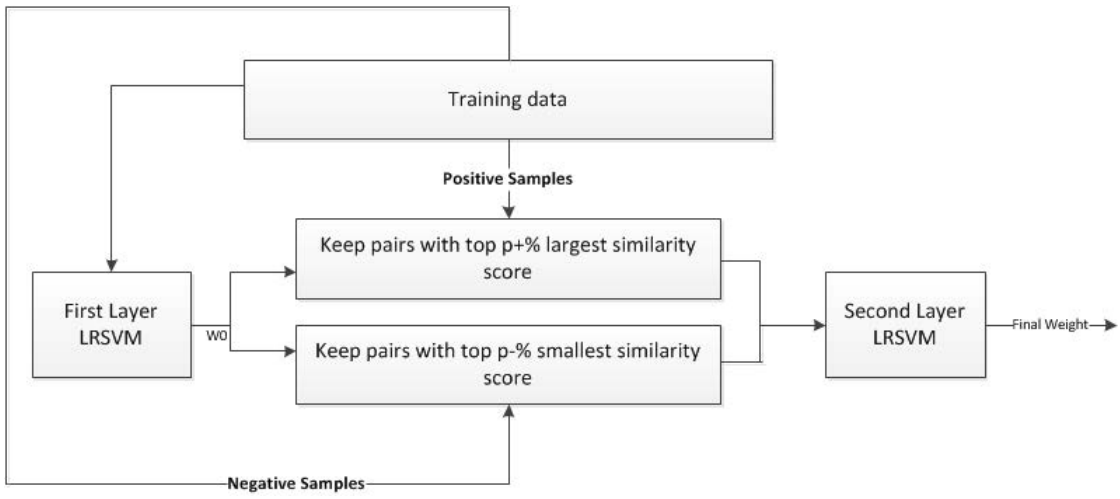

Figure 3.5: The Two-Layer LRSVM coupled with Pruning method.

Chapter 4

Experimental Results

To evaluate the performance of our proposed system, we conducted experiments on the highly challenging VIPeR dataset [2]. We provide the performance comparison between our system and several state-of-art methods based on the Cumulative Match Characteristic (CMC, see [54] for details), which can be seen as the *recall* at rank r. Additionally, some evaluations of the effectiveness of different feature channels are also provided.

4.1 Dataset and Settings

Although many existing datasets are available for pedestrian re-identification problem, there is really limited number of dataset designed for viewpoint invariant pedestrian retrieval. For example, in [55, 9], authors present results on pedestrian datasets which contain primarily frontal pedestrian images only. While this is reasonable in some scenarios such as a confined indoor camera network, many surveillance scenar-

ios require the ability to track pedestrians in large, open and uncontrolled weather conditional environments such as public plazas, campuses, communities and airport terminals. In these scenarios, a pedestrian may be seen from any angle from various cameras. This is the motivation for us to use VIPeR dataset to test our viewpoint and illumination invariant approach.

Viewpoint angle	0	45	90	135
45	16			
90	241	47		
135	43	72	4	
180	103	53	50	3

Table 4.1: The distribution of viewpoint angles in VIPeR dataset.

Viewpoint angle disparity	Examples
45	70
90	363
135	96
180	103

Table 4.2: The distribution of viewpoint angle changes in VIPeR dataset.

Table 4.1 lists the viewpoint statistic of images in VIPeR dataset, and we can see the large angle changes within the same individual image pairs in Table 4.2, where nearly 90% pairs have angle disparity of at least 90°. For details, all images are normalized to 128×48 pixels.

In our experiments, we run two protocols. In the first protocol, which is widely adopted by other work, we randomly select half (316) of the data for training and the rest (316) of the data for testing. While the latter use less (158) for training and more (474) for testing. We fully utilize the images in training set, group every possible unique pairs in this set, resulting one original positive pair, 104 inflated

positive pairs from duplicate samples and 315 negative pairs (take the first protocol as an example) for each probe image. While for testing, we generate each test set with a gallery set and a probe set. The probe set consists of one image for each person, and the remaining images are used as gallery set for ranking. Ideally rank 1 should be assigned only to the correct pair matches. However, the VIPeR dataset is the most challenging dataset thus made comparisons at ranks such as top 10 or top 20 more reasonable.

4.2 Evaluation of Effectiveness

To evaluate the effectiveness of the methods proposed in the previous sections, we conducted a series of experiments on various combinations of feature and distance learning methods. The details of configurations and explanations are listed in Table 4.3. We use the first protocol mentioned and run each test 10 times. In each

Configuration	Explanation
Baseline	Color Histograms Feature + Euclidean Distance
SSCD+	Baseline + SSCD
FV+	Baseline + FV
LBP+	Baseline + LBP
Feat+	Baseline + SSCD + FV + LBP
LRSVM	Proposed feature combination + Linear Ranking SVM
SISWD+	Proposed feature + LRSVM + SISWD
Proposed	Complete version

Table 4.3: The evaluation settings and explanation.

run, we randomly split the dataset into training and testing sets. The comparison of results is shown in Table 4.4. The cumulative matching characteristic (CMC) curves are presented in Figure 4.1.

Method($p = 316$)	$r = 1$	$r = 5$	$r = 10$	$r = 20$	$r = 50$
Baseline	0.3±0.3	4.7±0.4	9.3±0.6	15.4±0.6	22.9 ±0.7
SSCD+	2.1±0.3	11.4±0.5	19.4±0.6	25.8±0.6	32.1 ±0.6
FV+	5.4±0.3	18.6±0.4	27.9±0.6	34.3±0.7	43.6 ±0.8
Feat+	7.2±0.3	21.7±0.4	30.5±0.8	39.1±0.6	49.3 ±0.7
LRSVM	12.0±0.9	34.5±0.7	52.8±1.0	64.5±1.1	77.6 ±0.7
SISWD+	15.3±0.4	40.1±0.6	57.4±0.7	68.9±0.9	84.5 ±0.7
Proposed	**16.1±0.6**	**43.2±0.8**	**60.4±0.6**	**73.5±0.8**	**88.3 ±0.9**

Table 4.4: The evaluation results of different settings on VIPeR dataset with $p = 316$.

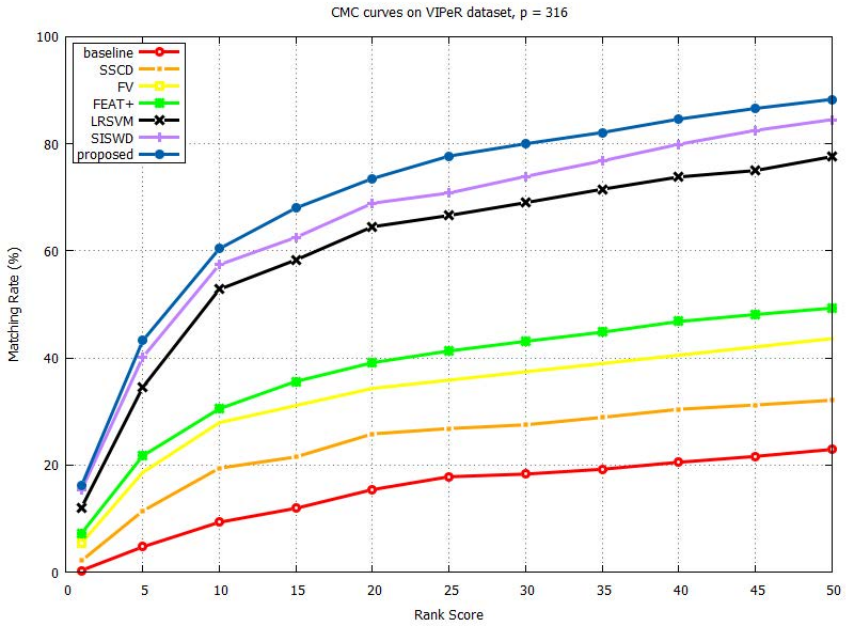

Figure 4.1: The effectiveness evaluation of our proposed methods. It is shown that our methods are all very effective in terms of performance.

We use a 10-fold cross-validation to search for the best pruning parameters p^+ and p^- given training dataset, as is plotted in Figure 4.2. We achieve the best performance when $p^+ = 0.9$ and $p^- = 0.85$, and they are used in the pruning process in two-layer LRSVM training.

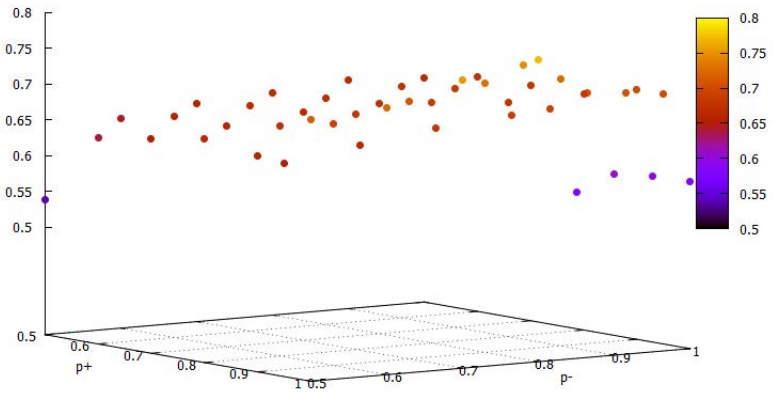

Figure 4.2: The pruning parameters p^+ and p^- cross-validation result.

Some good example results are shown in Figure 4.3, we sorted the top 10 images from left to right given the probe image on the left. The images marked with red boxes are the correct matches. We also collected some of the bad results with ranks more than 20, shown in Figure 4.4. It is worth noting that pedestrian with consistent bright color clothes can always achieve good ranks, while for pale colors such as white, black and gray, even human can hardly tell the difference, let alone the selected weight in feature space. Besides, we can observe some vital appearance change between two images of the same pedestrian, which made the dataset even more challenging.

4.3 Performance Comparison

There are several state-of-art results available on VIPeR dataset, however, some papers only provide CMC curves as figures, and many are using different ranking observations. Leave alone these massive inconsistent results, we stick to ($r = 1, r = 5, r = 10, r = 20$, and $r = 50$), and compare to [56, 42, 2, 57, 15, 58, 1, 44, 13] by using the results as best as we could achieve. The fairly comparative results under protocol 1 and 2 are shown in Table 4.5 and Table 4.6 respectively, note that only best run results are provided. Additionally, we marked the top 3 best results in Table 4.5 and top 1 result in Table 4.6. Compared to others, our proposed system can achieve good results especially at rank $r = 10$ and $r = 20$. At rank $r = 100$, we can obtain 98% retrieval rate which means given a probe image, the target image will appear in the candidate list of 100 with great confidence. Through analysis, we observe that some methods tend to achieve high absolute recognition rate such as SDALF, PS, RankBoost, and the miss rates are accordingly high. While other methods including ours, are better at higher ranks at 10 or 20. Although it is hard to conclude, one possible explanation is the penalty term to trade-off between recognition and retrieval rate are different. We could treat them as two approaches for different applications. For example, if we could keep track of a candidate list of 10, a higher retrieval rate is prior to high recognition rate. In contrast, when absolute target result is required, the higher recognition rate is better. The comparison indicates the power of kernel based learning techniques, we did not follow this scheme for now because the training time is a serious problem. According to [44], a training dataset that is much smaller than ours (see Section 3.1) took several days to run while our linear version only need less than half an hour, however, kernel based model is absolutely one of the directions

we will work on in the future.

The cumulative matching characteristic curves are plotted in Figure 4.5 and Figure 4.6 , respectively. Follow [54], we also evaluated the performance as a traditional

Method ($p = 316$)	$r = 1$	$r = 5$	$r = 10$	$r = 20$
ELF[2]	12.3	30.9	44.2	61.1
ITM[34]	11.6	31.4	45.8	63.9
MCC[59]	15.2	41.8	57.6	73.4
PRSVM[15]	14.7	36.4	50.8	66.8
PRDC[36]	15.7	38.4	53.9	70.1
SDALF[60]	**19.9**	38.9	49.4	65.7
PS[14]	**21.8**	**44.6**	57.2	71.2
RankBoost[1]	**23.9**	**45.6**	56.2	68.7
LMNN-R[56]	18.2	42.5	55.5	69.8
PCCA$-sqrt$[44]	17.3	42.4	**56.7**	**74.5**
PCCA$-rbf$[44]	19.3	**48.9**	**64.9**	**80.3**
Our method	16.8	44.0	**61.1**	**74.5**

Table 4.5: The comparison results of different methods with $p = 316$ testing images in the gallery.

Method ($p = 474$)	$r = 1$	$r = 5$	$r = 10$	$r = 20$
ELF	7.6	18.5	29.2	41.6
Bhat	5.3	14.1	21.6	31.7
L1-Norm	7.7	18.4	22.6	32.1
Ensemble-RankSVM	8.3	25.8	37.2	50.4
RankBoost	5.1	14.2	21.8	32.9
PRSVM	**9.1**	25.9	39.4	51.2
Proposed	8.9	**28.3**	**42.6**	**54.4**

Table 4.6: The comparison results of different methods given $p = 474$ testing images in the gallery.

recognition problem. Assume that a set of M pedestrian that enter a camera network are *i.i.d.* Samples from some large testing dataset of size N. If these M pedestrians cross from one camera to another at the same time we have a reacquisition problem

where we must find the correct matching configuration. If the CMC curve for the matching function is given, we can calculate the probability that any of the M best matches is correct as follows:

$$SDR(M) = CMC(N/M) \tag{4.1}$$

where $CMC(k)$ is the rank k recognition rate.

We again compared the recognition performance with available results, shown in Figure 4.7. Our system is able to outperform other work except the kernel based PCCA method.

Figure 4.3: Some good results achieved.Left: Probe image. Middle: Top 10 results sorted from left to right. The images with red boxes represent the correct matches.

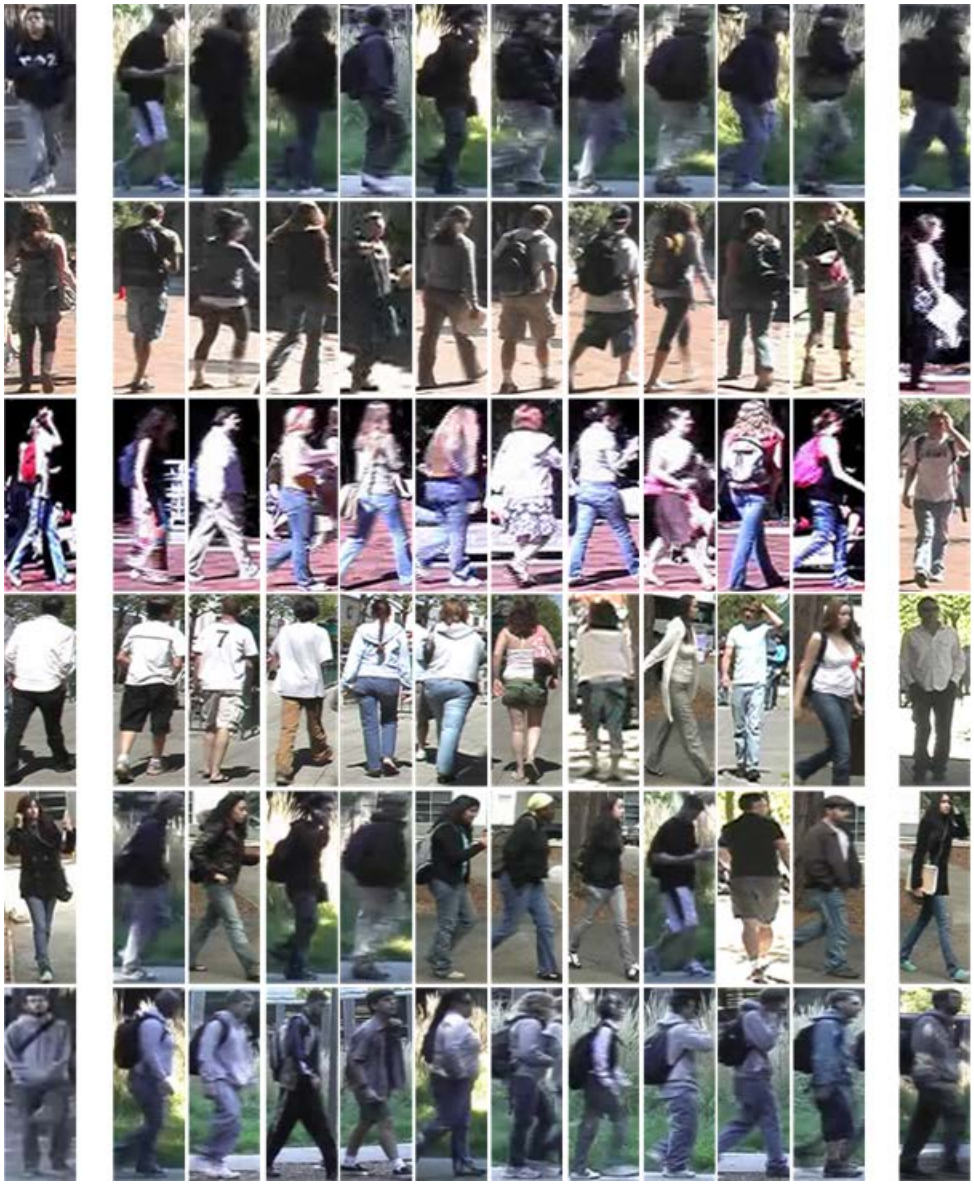

Figure 4.4: The example set of results that are not ideal with ranks $r > 20$.

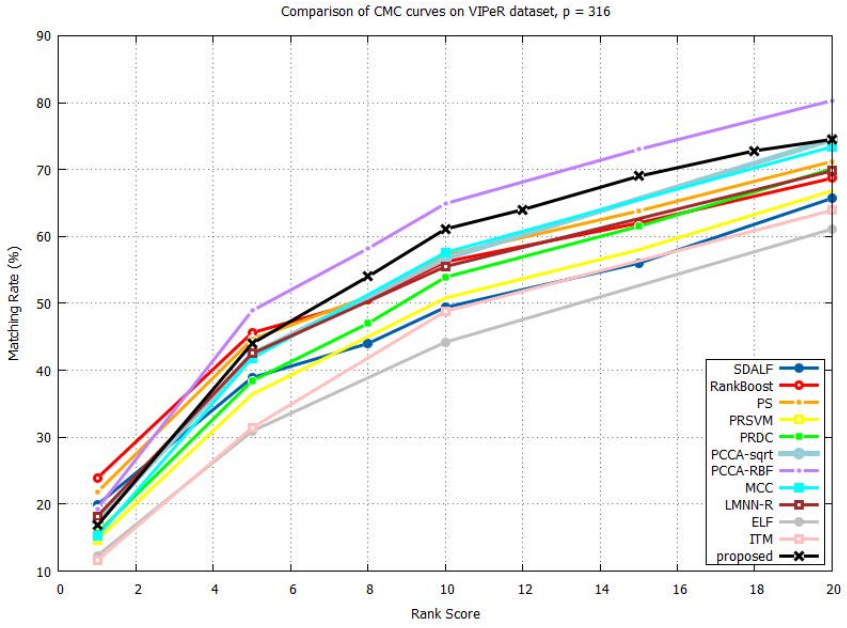

Figure 4.5: Performance comparison using CMC curves as the function or r, given $p = 316$ test samples.

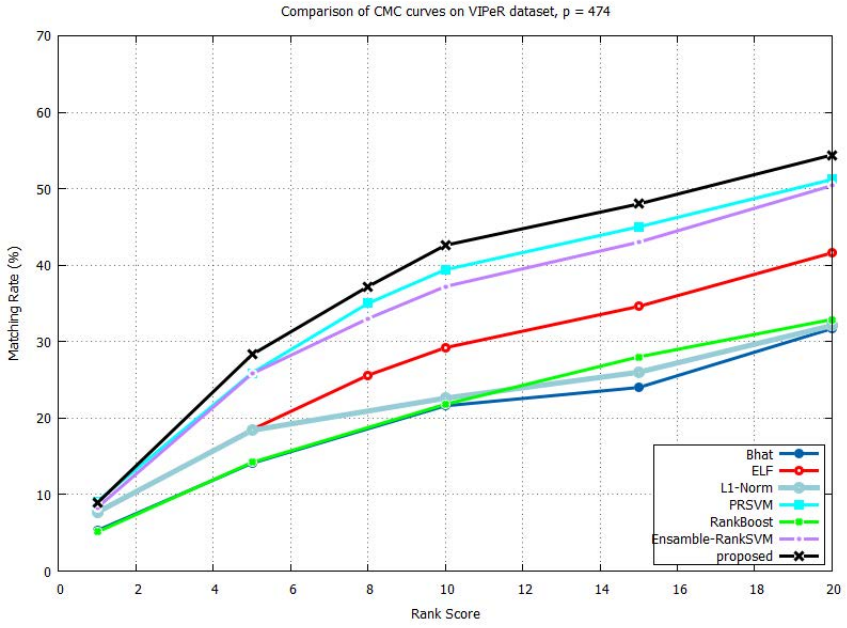

Figure 4.6: Performance comparison using CMC curves as the function or r, given $p = 474$ test samples.

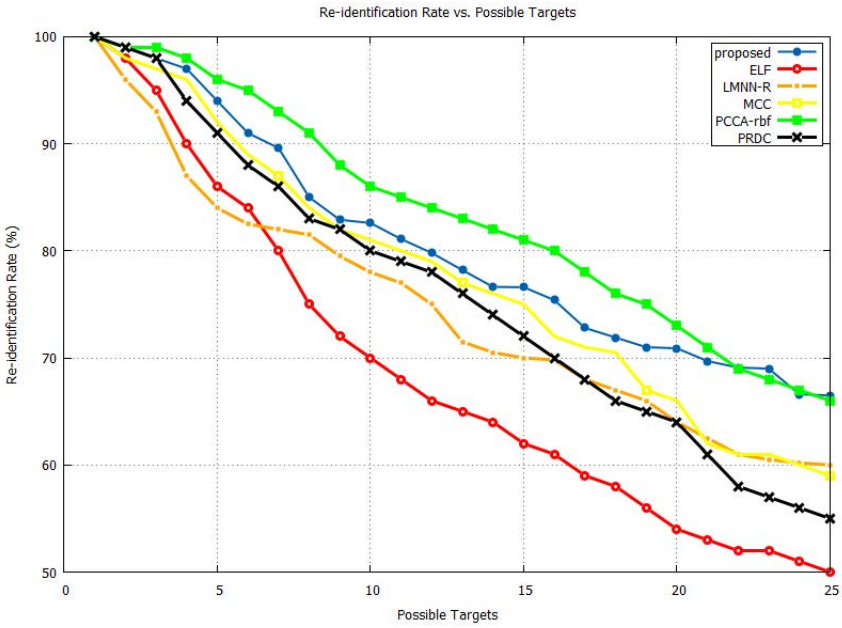

Figure 4.7: The comparison of Re-identification Rate *vs.* Possible Targets.

Chapter 5

Summary and Concluding Remarks

We have proposed a method for person re-identification that can be generally divided into two parts: feature and learning. In the first part, the proposed framework take advantage of low-level details such as multiple color channels, semantic color names, LBP and gradients, as well as global information including histograms of low-level features, statistics of color names and Fisher vectors encoded local descriptors, to provide discriminative image appearance representation. In terms of distance learning, unlike existing methods, we utilize as much information from pairs of images as possible, and provide a two-layer linear ranking SVM to handle the large-scale training data. A SISWD method is applied in this process to deal with the imbalanced data. We also developed a pruning method for better discrimination analysis.

Quantitative evaluation results have shown the effectiveness of our system which provides encouraging performance on the most challenging pedestrian re-identification dataset as some state-of-art work. And it is even satisfying if we take our training module execution time into account which is way less than the kernel based learning

methods.

However, we still focus on certain aspects of our work to improve the current performance, such as a better combination of feature representation, multi-layer weight control rather than fixed two-layer scheme and speedup for kernel based learning.

Bibliography

[1] Cheng-Hao Kuo, Sameh Khamis, and Vinay Shet. Person re-identification using semantic color names and rankboost. In *WACV*, pages 281–287. Citeseer, 2013.

[2] Douglas Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *ECCV 08: Proceedings of the 10th European Conference on Computer Vision*, pages 262–275, Berlin, Heidelberg, 2008. Springer-Verlag, Springer-Verlag.

[3] Christopher Madden, Eric Dahai Cheng, and Massimo Piccardi. Tracking people across disjoint camera views by an illumination-tolerant appearance representation. *Machine Vision and Applications*, 18(3-4):233–247, 2007.

[4] Martin Hirzer, Csaba Beleznai, Peter M Roth, and Horst Bischof. Person re-identification by descriptive and discriminative classification. In *Image Analysis*, pages 91–102. Springer, 2011.

[5] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[6] Omar Oreifej, Ramin Mehran, and Mubarak Shah. Human identity recognition in aerial images. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 709–716. IEEE, 2010.

[7] William Robson Schwartz and Larry S Davis. Learning discriminative appearance-based models using partial least squares. In *Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on*, pages 322–329. IEEE, 2009.

[8] Slawomir Bak, Etienne Corvee, Francois Brémond, and Monique Thonnat. Person re-identification using spatial covariance regions of human body parts. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 435–440. IEEE, 2010.

[9] Niloofar Gheissari, Thomas B Sebastian, and Richard Hartley. Person reidentification using spatiotemporal appearance. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1528–1535. IEEE, 2006.

[10] Wei shi Zheng, Shaogang Gong, and Tao Xiang. Associating groups of people. In *BMVC*, 2009.

[11] Kai Jungling, Christoph Bodensteiner, and Michael Arens. Person re-identification in multi-camera networks. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 55–61. IEEE, 2011.

[12] P-E Forssén. Maximally stable colour regions for recognition and matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[13] Michela Farenzena, Loris Bazzani, Alessandro Perina, Vittorio Murino, and Marco Cristani. Person re-identification by symmetry-driven accumulation of local features. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2360–2367. IEEE, 2010.

[14] Dong Seon Cheng, Marco Cristani, Michele Stoppa, Loris Bazzani, and Vittorio Murino. Custom pictorial structures for re-identification. In *BMVC*, volume 2, page 6, 2011.

[15] Bryan Prosser, Wei-Shi Zheng, Shaogang Gong, Tao Xiang, and Q Mary. Person re-identification by support vector ranking. In *BMVC*, volume 1, page 5, 2010.

[16] Ying Zhang and Shutao Li. Gabor-lbp based region covariance descriptor for person re-identification. In *Image and Graphics (ICIG), 2011 Sixth International Conference on*, pages 368–371. IEEE, 2011.

[17] Bryan Prosser, Shaogang Gong, Tao Xiang, et al. Multi-camera matching under illumination change over time. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications-M2SFA2 2008*, 2008.

[18] Li Fei-Fei, Rob Fergus, and Antonio Torralba. Recognizing and learning object categories. *CVPR Short Course*, 2, 2007.

[19] Florent Perronnin and Christopher Dance. Fisher kernels on visual vocabularies for image categorization. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[20] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *Computer Vision–ECCV 2010*, pages 143–156. Springer, 2010.

[21] Florent Perronnin, Yan Liu, Jorge Sánchez, and Hervé Poirier. Large-scale image retrieval with compressed fisher vectors. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3384–3391. IEEE, 2010.

[22] Ken Chatfield, Victor Lempitsky, Andrea Vedaldi, and Andrew Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. 2011.

[23] Michael Hahnel, Daniel Klunder, and K-F Kraiss. Color and texture features for person recognition. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, volume 1. IEEE, 2004.

[24] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

[25] Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *The Journal of machine learning research*, 4:933–969, 2003.

[26] Olivier Chapelle and S Sathiya Keerthi. Efficient algorithms for ranking with svms. *Information Retrieval*, 13(3):201–215, 2010.

[27] Martin Hirzer, Peter M Roth, Martin Köstinger, and Horst Bischof. Relaxed pairwise learned metric for person re-identification. In *Computer Vision–ECCV 2012*, pages 780–793. Springer, 2012.

[28] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55, 1936.

[29] M Kostinger, Martin Hirzer, Paul Wohlhart, Peter M Roth, and Horst Bischof. Large scale metric learning from equivalence constraints. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2288–2295. IEEE, 2012.

[30] Aharon Bar-Hillel, Tomer Hertz, Noam Shental, Daphna Weinshall, and Greg Ridgeway. Learning a mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research*, 6(6), 2005.

[31] Gal Chechik, Varun Sharma, Uri Shalit, and Samy Bengio. Large scale online learning of image similarity through ranking. *The Journal of Machine Learning Research*, 11:1109–1135, 2010.

[32] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Person re-identification by probabilistic relative distance comparison. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 649–656. IEEE, 2011.

[33] Matthieu Guillaumin, Jakob Verbeek, and Cordelia Schmid. Is that you? metric learning approaches for face identification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 498–505. IEEE, 2009.

[34] Jason V Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007.

[35] Brent Berlin. *Basic color terms: Their universality and evolution.* Univ of California Press, 1991.

[36] Joost Van De Weijer and Cordelia Schmid. Applying color names to image description. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, volume 3, pages III–493. IEEE, 2007.

[37] Arthur P Dempster, Nan M Laird, Donald B Rubin, et al. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal statistical Society*, 39(1):1–38, 1977.

[38] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):971–987, 2002.

[39] Matti Pietikäinen, Timo Ojala, and Zelin Xu. Rotation-invariant texture classification using feature distributions. *Pattern Recognition*, 33(1):43–52, 2000.

[40] Mäenpää Topi, Ojala Timo, Pietikäinen Matti, and Soriano Maricor. Robust texture classification by subsets of local binary patterns. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 3, pages 935–938. IEEE, 2000.

[41] Chi-Ho Chan, Josef Kittler, and Kieron Messer. *Multi-scale local binary pattern histograms for face recognition.* Springer, 2007.

[42] Bingpeng Ma, Yu Su, and Frédéric Jurie. Local descriptors encoded by fisher vectors for person re-identification. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 413–422. Springer, 2012.

[43] Sateesh Pedagadi, James Orwell, Sergio Velastin, and Boghos Boghossian. Local fisher discriminant analysis for pedestrian re-identification. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3318–3325. IEEE, 2013.

[44] Alexis Mignon and Frédéric Jurie. Pcca: A new approach for distance learning from sparse pairwise constraints. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2666–2672. IEEE, 2012.

[45] Olivier Chapelle and Yi Chang. Yahoo! learning to rank challenge overview. *Journal of Machine Learning Research-Proceedings Track*, 14:1–24, 2011.

[46] Ralf Herbrich, Thore Graepel, and Klaus Obermayer. Large margin rank boundaries for ordinal regression. *Advances in Neural Information Processing Systems*, pages 115–132, 1999.

[47] Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11:23–581, 2010.

[48] Zhaohui Zheng, Keke Chen, Gordon Sun, and Hongyuan Zha. A regression framework for learning ranking functions using relative relevance judgments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 287–294. ACM, 2007.

[49] Ching-Pei Lee and Chih-Jen Lin. Large-scale linear ranksvm. *Neural Computation*, pages 1–37, 2014.

[50] Andrew R Conn, Nicholas IM Gould, and Ph L Toint. *Trust region methods*, volume 1. Siam, 2000.

[51] Chih-Jen Lin and Jorge J Moré. Newton's method for large bound-constrained optimization problems. *SIAM Journal on Optimization*, 9(4):1100–1127, 1999.

[52] Chih-Jen Lin, Ruby C Weng, and S Sathiya Keerthi. Trust region newton method for logistic regression. *The Journal of Machine Learning Research*, 9:627–650, 2008.

[53] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, 2008.

[54] Douglas Gray, Shane Brennan, and Hai Tao. Evaluating appearance models for recognition, reacquisition, and tracking. In *IEEE International workshop on performance evaluation of tracking and surveillance*. Citeseer, 2007.

[55] Chris Stauffer and Eric Grimson. Similarity templates for detection and recognition. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–221. IEEE, 2001.

[56] Mert Dikmen, Emre Akbas, Thomas S Huang, and Narendra Ahuja. Pedestrian recognition with a learned metric. In *Computer Vision–ACCV 2010*, pages 501–512. Springer, 2011.

[57] Wei Li, Rui Zhao, and Xiaogang Wang. Human reidentification with transferred metric learning. In *Computer Vision–ACCV 2012*, pages 31–44. Springer, 2013.

[58] Le An, Mehran Kafai, Songfan Yang, and Bir Bhanu. Reference-based person re-identification. In *Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on*, pages 244–249. IEEE, 2013.

[59] Amir Globerson and Sam Roweis. Metric learning by collapsing classes. In *Nips*, volume 18, pages 451–458, 2005.

[60] Rudolf Bayer. Symmetric binary b-trees: Data structure and maintenance algorithms. *Acta informatica*, 1(4):290–306, 1972.