The Empirical Moment Matrix and Its Application in Computer Vision

A Dissertation Presented
by

Mengran Gou

to

The Department of Electrical and Computer Engineering

in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in

Electrical Engineering

Northeastern University
Boston, Massachusetts

April 2018

To my family and parents.

Contents

List of Figures

List of Tables

List of Acronyms

ACF Aggregated Channel Features

BBox Bounding boxes

BCNN Bilinear Convolution Neural Network

BoW Bag of Words

cam camera

CNN Convolutional Neural Network

DPM Deformable Parts Model

DynF Dynamic-based feature

DynMo Dynamic-based Moment encoding feature

DynFV Dynamic-based Fisher Vector encoding feature

FDA Fisher discriminant analysis

GMM Gaussian Mixture Model

GOG Gaussian of Gaussian descriptor

HOG3D Histogram Of Gradient 3D

KISSME Keep-It-Simple-and-Straightforward MEtric

LDFV Local Descriptors encoded by Fisher Vector

LE Log-Euclidean Riemannian Metric

LFDA Local Fisher Discriminant Analysis

LOMO Local Maximal Occurrence feature

MFA Marginal Fisher Analysis

moM Mean of Moment matrix feature

moMaGO Combination of Mean of Moment matrix feature and Gaussian of Gaussian feature

MoNet Moment Embedding Network

NFST Discriminative Null Space Learning

PCCA Pairwise Constrained Component Analysise

re-ID person Re-Identification

SDC Dense Color SIFT feature

SPD symmetric positive definite

WHOS Weighted Histograms of Overlapping Stripes feature

XQDA cross-view quadratic discriminant analysis

Acknowledgments

First and foremost, I would like to thank my advisor, Professor Octavia Camps for her patience and support through my entire Ph.D. life. Her shoulder-by-shoulder guidance set a solid problem-solving foundation for my research building. Also, the wide variety of projects she provided granted me opportunities to apply my knowledge in computer vision on real-world applications, which motivates me to keep thinking and breaking my limits. Her rigorous academic value conveys to me a lifetime benefit.

I would also like to thank Professor Mario Sznaier for his guidance and help on all kinds of different fundamental theoretical problems. His endless support makes me feel worried less to pursue the highest goal. I also want to thank Professor Richard Radke for showing me how to collaborate with other team players and the careful counsel on my manuscripts.

I would like to thank Professor Jennifer Dy for serving as my committee member and for her insightful comments on my work.

I would like to thank Dr. Milind Rajadhyaksha and Dr. Kivanc Kose for their help and wise advice on medical image processing. I'm thrilled to see my knowledge can help cancer patients.

I also want to thank Professor Dana Brooks, Professor Mark Niedre and Professor Ningfang Mi for their patience and thoughtful guidance on the organization of NEPSSS seminars. I enjoyed the time with NEPSSS!

I would like to thank the supportive faculties and staff in ALERT, especially Professor Michael Silevitch for his insightful advice and the "so what who cares" and Deanna Beirne for her help to maintain the machines and equipment of the project.

I want to thank my collaborators from RPI, Dr. Ziyan Wu, Dr. Yang Li and Dr. Srikrishna Karanam for their thoughtful discussions and help on the CLE Airport project.

I spent my six year Ph.D. life in one of the best labs, and I would thank all my lab mates, collaborators, mentors and best friends in the Robust System Lab. I would like to thank Binlong, Fei, Walter, Xuefeng, Xiao, Burak, Tom, Oliver, Yongfang, Yin, Ichiro, Zulqarnain, Jing, Weijia, Caglayan, Rasmus, Jose, Sadjad, Tianyu, Wenqian, Bengisu, Angels, Qian, Begum, Xikang, Dong and Yuexi for their sharing and support.

Last but the most important, I would like to thank three unique women in my life, my wife Dr. Wenjuan Qin and my daughters Shuhan and Shuxin. They are my source of happiness and power to help and support me through the tough but exciting six years.

Notations

$\mathbb{R}, \mathbb{N}$	set of real number, set of nonnegative integers
$x, \mathbf{x}, \mathbf{X}$	scalar, a vector in $\mathbb{R}^n$, a matrix in $\mathbb{R}^{m\times n}$
$\mathbf{x}(i)$	the i-th entry of $\mathbf{x}$
$\mathbf{X}(i,j)$	the (i,j)-th entry of $\mathbf{X}$
$\\|\mathbf{X}\\|_F$	Frobenius norm of the matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$ $\\|\mathbf{X}\\|_F \doteq \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbf{X}(i,j)^2}$
$\\|\mathbf{x}\\|_2$	ℓ_2-norm of the vector $\mathbf{x} \in \mathbb{R}^n$ $\\|\mathbf{x}\\|_2 \doteq \sqrt{\sum_{i=1}^{n}\mathbf{x}(i)^2}$
$\\|\mathbf{x}\\|_1$	ℓ_1-norm of the vector $\mathbf{x} \in \mathbb{R}^n$ $\\|\mathbf{x}\\|_1 \doteq \displaystyle\sum_{i}\lvert x_i\rvert$
$s_{m,D}$	$\displaystyle\binom{m+D}{m}$

Abstract of the Dissertation

The Empirical Moment Matrix and Its Application in Computer Vision

by

Mengran Gou

Doctor of Philosophy in Electrical Engineering

Northeastern University, April 2018

Dr. Octavia I. Camps, Advisor

Embedding local properties of an image, for instance its color intensities or the magnitude and orientation of its gradients, to create a representative feature is a critical component in many computer vision tasks, such as detection, classification, segmentation and tracking. A feature that is representative yet invariant to nuisance factors will scaffold the following modules in the processing pipeline and lead to a better performance for the task at hand. Statistical moments have often been utilized to build such descriptors since they provide a quantitative measure for the shape of the underlying distribution of the data. Examples of these include the covariance matrix feature, bilinear pooling encoding and Gaussian descriptors. However, until now, these features have been limited to using up to second order moments, i.e. the mean and variance of the data, and hence can be poor descriptors when the underlying distribution is non-Gaussian. This dissertation aims towards examining this problem in-depth and identifying possible solutions. In particular, we propose to use feature descriptors based on the empirical moment matrix, which gathers high order moments and embeds them into the manifold of symmetric positive definite (SPD) matrices. The effectiveness of the proposed approach is illustrated in the context of two computer vision problems: person re-Identification (re-ID) and fine-grain classification.

Person re-ID is the problem of matching images of a pedestrian across cameras with no overlapping fields of view. It is one of the key tasks in surveillance video processing. Yet, due to the extremely large inter-class variances across different cameras (e.g., poses, illumination, viewpoints), the performance of the state-of-the-art person re-id algorithms is still far from ideal. In this thesis, we propose a novel descriptor, based on the on-manifold mean of a moment matrix (moM) and horizontal mean pooling, which can be used to approximate complex, non-Gaussian, distributions of the pixel features within a mid-sized local patch. To mitigate the gap between academic research and real-world applications, two large-scale public re-ID datasets are proposed and a systematic benchmark evaluation is established on both new datasets. Extensive experiments on five widely used public re-ID datasets and two newly collected datasets demonstrate that incorporating the proposed moM feature improves re-ID performance.

Different from general objection recognition tasks, fine-grained classification usually tries to distinguish objects at the sub-category level, such as different makes of cars or different species of a bird. The main challenge of this task is the relatively large inter-class and relatively small intra-class variations. The most successful approaches to this problem use deep convolutional neural network(CNN), where the top convolutional layers perform a local representation extraction step and the bottom fully connected layers perform an encoding step. In the case of fine-grain classification, bilinear pooling and Gaussian embedding have been shown as the best encoding options but at the price of an enormous feature dimensionality. Approximate compact pooling methods have been explored towards addressing this weakness. Additionally, recent results have shown that significant performance gains can be achieved by using matrix normalization to regularize the unstable higher order information. However, combining compact pooling with matrix normalization has not been explored until now. In this thesis, we unify the bilinear pooling layer and the global Gaussian embedding layer through the empirical moment matrix in a novel deep architecture, moment embedding network MoNet. In addition, we propose a novel sub-matrix square-root layer, which can be used to normalize the output of the convolution layer directly and mitigate the dimensionality problem with off-the-shelf compact pooling methods. Our experiments on three widely used fine-grained classification datasets illustrate that our proposed architecture MoNet can achieve similar or better performance than the state-of-art architectures . Furthermore, when combined with compact pooling techniques, it obtains comparable performance with encoded features but with only 4% of the dimensions.

Chapter 1

Introduction

In most computer vision tasks, embedding local representations of an image to form a feature that is representative yet invariant to nuisance factors is a crucial step. Essentially, this procedure amounts to estimating the underlying distribution of the data and summarizing it into some compact representation. In statistics, a moment is a quantitative measure of the appearance of a distribution. For example, the first order moment represents the mean value and the second order moment represents the variance. Thus, a collection of different order moments can be utilized to approximate the distribution of the data, where using the higher order of the moments leads to the more accurate representation. However, the current statistical features have been limited to using up to second order moments, i.e., the mean and the variance, which is insufficient when the underlying distribution is non-Gaussian. This research aims towards examining this problem in-depth and identifying possible solutions.

In this dissertation, the use of the empirical moment matrix, a unique formulation of the collection of moments, is proposed as the way to capture local statistics. The main advantage of using the moment matrix is that, in addition to gathering 0 to high order moments, it provides a natural embedding of the data into the manifold of (SPD) matrices, which does not depend on the number of samples. The benefits of using such embedding will be illustrated in the context of two challenging computer vision problems: person (re-ID) and fine-grained classification.

1.1 Challenges

Person re-ID is the problem of matching images of a pedestrian across cameras with no overlapping fields of view. It is one of the critical tasks in surveillance video processing. Yet, due to the extremely large inter-class variances across different cameras (e.g., poses, illumination, viewpoints), the performance of the state-of-the-art person re-id algorithms is still far from ideal.

Different from general objection recognition tasks, fine-grained classification usually tries to distinguish objects at the sub-category level, such as different makes of cars or different species of birds. The main challenge of this task is the relatively large inter-class and relatively small intra-class variations.

The use of statistical moments has been investigated to address many computer vision problems. For example, the covariance matrix has been proposed as a feature to describe an image or region [3, 4]. By taking advantage of both mean and covariance information, several a Gaussian descriptors have been proposed [5, 6, 7]. These approaches differ in how they inject the mean and covariance information. The Gaussian descriptors have a critical problem of assuming the underlying distribution is Gaussian, which is often not true. In order to approximate arbitrary distributions, Gaussian Mixture Model (GMM) [8] and its derivative extension, the Fisher Vector [9], have been proposed. However, the number of Gaussians in the mixture is a key hyper parameter and the estimation is usually a heuristic procedure.

In contrast, by computing the tensor product of a vector of ordered monomials, the proposed approaches using empirical moment matrix can approximate arbitrary distributions by including higher order moments.

1.2 Contributions

The main contributions of this dissertation are:

- A local feature encoding paradigm based on the empirical moment matrix that unifies and generalizes bilinear pooling and Gaussian descriptors with the ability to incorporate higher order moment information. Synthetic experiments illustrate the benefits of introducing higher order moments to model non-Gaussian distributions.

- A hierarchical hand-crafted feature (moM) for the person re-ID problem to better model non-Gaussian data. Empirical moment matrices are used to model the non-Gaussian local patches while the mean on the manifold of SPD matrices for the moment matrices of patches at the same height provides viewpoint invariance. Experiments on five widely used public datasets show the effectiveness of moM, which achieves comparable or better performances than state-of-the-art in all five datasets when combining with GOG [10].

- Two new real-world large-scale person re-ID datasets. One was captured with a surveillance camera network inside the sterile zone of an US Airport. The other was captured with an existing surveillance camera network at Duke University. A systematic benchmark evaluation with 10 features and 12 metric learning methods verifies that the proposed moM is complementary to GOG [10] and that the combination of both features achieves the best result.

- A mathematical formulation that uses the empirical moment matrix to disentangle the bilinear pooling layer from the global Gaussian embedding layer in CNN. Based on this result, together with a novel sub-matrix square-root layer, we proposed a new architecture, MoNet, which can take the advantages from both the Gaussian embedding and the matrix normalization.

- Experiments for the fine-grained classification problem show that the proposed MoNet architecture achieves similar or better results than G^2DeNet[11] and the Tensor Sketch version can achieve comparable performance with only 4% of its dimensionality.

The dissertation is organized as follows. After the introduction, Chapter 2 reviews existing work on feature encoding for both conventional and modern CNN pipelines. Then Chapter 3 introduces the proposed moM feature, which is followed by a chapter to discuss the experimental results on person re-ID. Chapter 6 describes the MoNet in detail, including the derivation of backward propagation. Experiments on fine-grained classification are discussed in Chapter 7 and the dissertation concludes with a discussion of the contributions, limitations and possible directions for future research.

Chapter 2

Image representations in computer vision

Aligning with the human visual system, computer vision cares about analysis and understanding the information from images or videos [12]. To achieve this, one essential and critical step is describing the image with a feature that is representative yet invariant to nuisance noise. In the past decades, a significant amount of work has been proposed on this topic [13, 14]. Most of existing image representation pipelines have two critical steps: *feature extraction* and *encoding* (Fig. 2.1). Given an image, a feature exaction step will extract local features to describe a small region, and the follow-up encoding step will aggregate those local features into one final representation vector. Recently, CNNs has been widely applied in different computer vision tasks and achieved a terrific success in most of them. Although CNNs are trained from end to end, the convolutional layers can be viewed as local feature extractors and the following fully connection (FC) layers as feature encoding steps.

This chapter will give a brief review of existing image representation techniques, in both conventional computer vision pipelines and modern deep learning pipelines.

2.1 Conventional pipeline

2.1.1 Feature Extraction

Given an image, it will provide color/gray intensity features naturally. With this information, one can split the hue, saturation and lightness with different color spaces for the color image. By

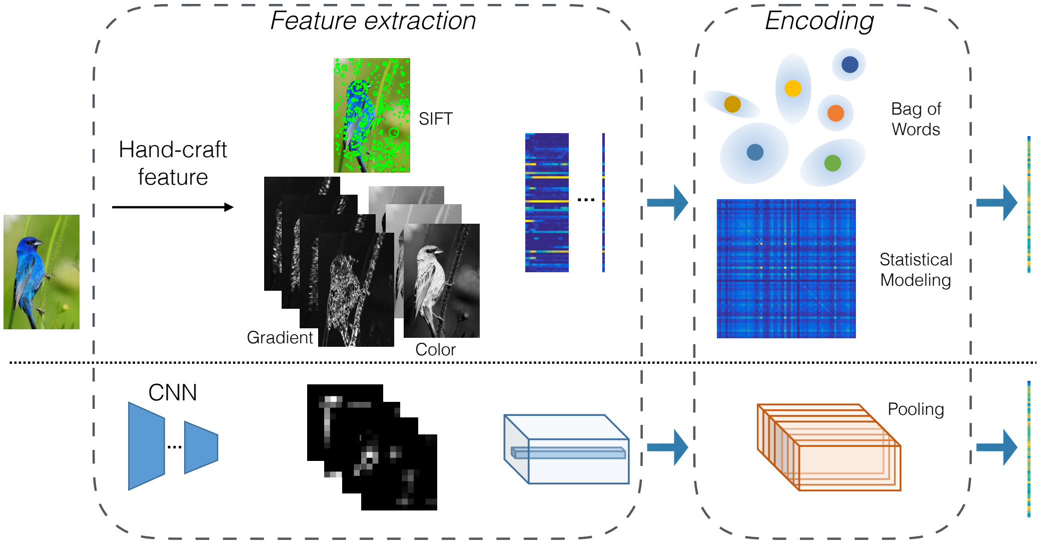


Figure 2.1: Typical image representation pipeline in a computer vision system. The upper rows shows a conventional pipeline whereas the lower row shows a modern deep CNN pipeline.

applying Gaussian derivatives in different orientations, one can obtain the edge or texture information for each pixel as well. Later, instead of using the local features for all pixels, key-points based local regions have been shown to be more robust to spatial variations. Harris corners [15] have been widely used to obtain rotation invariant features. To make the local feature more representative, several carefully designed craft local descriptors have been proposed. Scale-invariant feature transform (SIFT) [16] characterizes the local region with the histogram of gradient in sub-regions. For each local region, it uses the difference of Gaussians (DoG) to find the maximum scale and assigns the direction with the maximum magnitude of gradient as the orientation. With proper normalization, it achieves an exceptional performance on scale and orientation invariant representations. Speed up robust feature (SURF) [17] approximates the DoG with a box convolution filter and utilizes the response of wavelets to represent the local region. It can achieve comparable performance to SIFT but in a much faster way. In 2011, Rublee *et al.* proposed Oriented FAST and Rotated BRIEF (ORB) [18] as an efficient alternative for SIFT or SURF. By taking advantage of a much faster key-point detector FAST [19] and an efficient orientation normalization, it has been widely applied in many real-time applications.

2.1.2 Encoding

Most encoding methods in the literature can be classified into two main types. On one hand, adopted from document classification, Bag of Words (BoW) [20] clusters the local descriptors into a vocabulary set and model the whole image by counting the occurrences of local visual features. Instead of only counting the occurrences, VLAD [21] also keeps the difference between the local feature and the visual words. Fisher vector [9] adopts the GMM as the visual vocabulary and utilizes the derivative on the means and variances as the feature. On the other hand, statistical moments have also been used as a representation for a region of interest because they can estimate the shape of the underline distribution. Covariance matrix has shown remarkable results in texture classification [3], tracking [22] and person detection [23]. Bilinear pooling [24] uses the outer product of the feature vector to compute the second order statistical information to model two-factor structure image. Later, several formulations of Gaussian descriptors have been proposed to improve the performance by adding the mean information. [5] form the multivariate Gaussian with a positive definite lower triangular affine transformation matrix. [10] embeds it with a semi-positive definition (SPD) matrix and applied on re-ID.

2.2 Modern CNN pipeline

In 2012, Krizhevsky *et al.* [25] demonstrated the power of deep convolutional neural networks for computer vision tasks. With the help of modern GPU, and large-scale datasets, AlexNet pushed the best performance on an image classification challenge by a significant margin. After that, many works utilize pre-trained deep CNNs as local representation extractors or the final feature extractors to describe the image for different tasks. Figure 2.2 illustrates a typical modern CNN architecture, which includes a few convolutional layers, followed by several fully connected layers. One can easily align it with the conventional feature extraction pipeline by considering the convolutional layers as the local representation extractor and the fully connection layers as the encoding method. Similar to the conventional pipeline research path, intensive research has been done on both, convolutional layers as well as the layers right after that. After AlexNet, Simonyan *et al.* [2] substituted the large convolution filter with consecutive smaller ones and increased the number of layers. The proposed VGGNet improved the performance. He *et al.* [26] mitigated the gradient vanishing problem with a jump connection between non-connected convolution layers and increased the number of layers dramatically. The proposed network is named ResNet. Besides fully

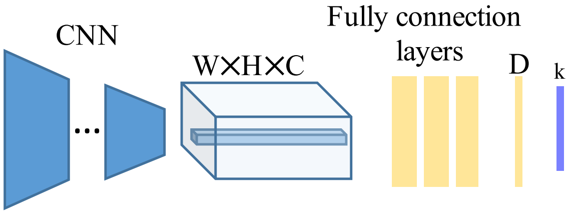


Figure 2.2: A typical modern deep CNN architecture

connection layer, global average pooling [26] has been used to extract the final feature for the image. Inspired by [27], Lin *et al.* [28] proposed to substitute the FC layers with a bilinear pooling layer, which has been proven to be effective in fine-grained classification [28], texture analysis [29] and segmentation [30]. A critical drawback for bilinear pooling is the large dimensionality due to the outer product. To mitigate this problem, Gao *et al.* [31] applied random Maclaurin [32] and tensor sketch [33] techniques to approximate the bilinear pooling layer with much less dimensionality. In [34], the low-rank property was investigated to reduce the number of parameters for the network. Wang *et al.* [11] extended bilinear pooling with global Gaussian embedding layer. Also, a proper matrix square-root normalization technique was also discussed. Around the same time, the author of BCNN discovered the similar matrix power normalization scheme and named it as improved BCNN [35].

Chapter 3

moM: mean of Moments Feature

In the past decade, different types of descriptors have been proposed and tested on the re-id problem. Two recently proposed techniques led to a significant improvement in the quality of these descriptors [36, 10].

The first technique replaces the simple computation of histograms with more advanced feature-encoding methods. Along this line, covariance matrices are used to encapsulate the second order moment information in a local patch [37, 38]. By recognizing the importance of also including the first order moment in the feature representation, [39, 10] achieved the state-of-the-art performance using a SPD embedded Gaussian descriptor. However, a limitation of this descriptor is the implicit assumption that the underlying distribution is a Gaussian. When this assumption does not hold, (see Figure 3.1), up to second order moment information is not sufficient to completely represent relatively complex local regions. Though Fisher Vector encoding feature can mimic a non-Gaussian distribution with GMM and achieve decent results on re-id [40, 41], it assumes that the variables at the pixel-level feature are independent from each other. Moreover, the GMM needs a training set to learn its parameters. In contrast, here we propose to take into account higher (greater than two) order moment information by using the empirical moment matrix to approximate arbitrary non-Gaussian distributions in the local region without requiring learning parameters.

The second technique applies a strip level pooling step to further improve cross-view invariance. As identities are roughly aligned in the vertical direction (Figure 3.1), different viewpoints would mainly affect the appearance distribution in the horizontal direction. Based on this assumption, Liao *et. al* [42] apply maximum pooling along the same height and Matsukawa *et. al* [10] uses another Gaussian model to approximate the distribution of the dense patches descriptors. In this dissertation, we also use horizontal mean pooling to improve the feature viewpoint invariance.

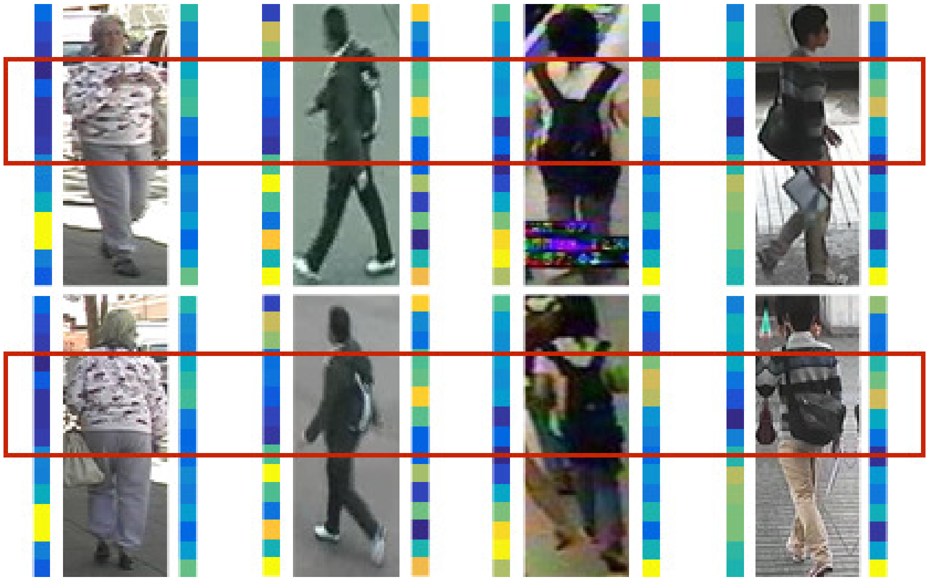
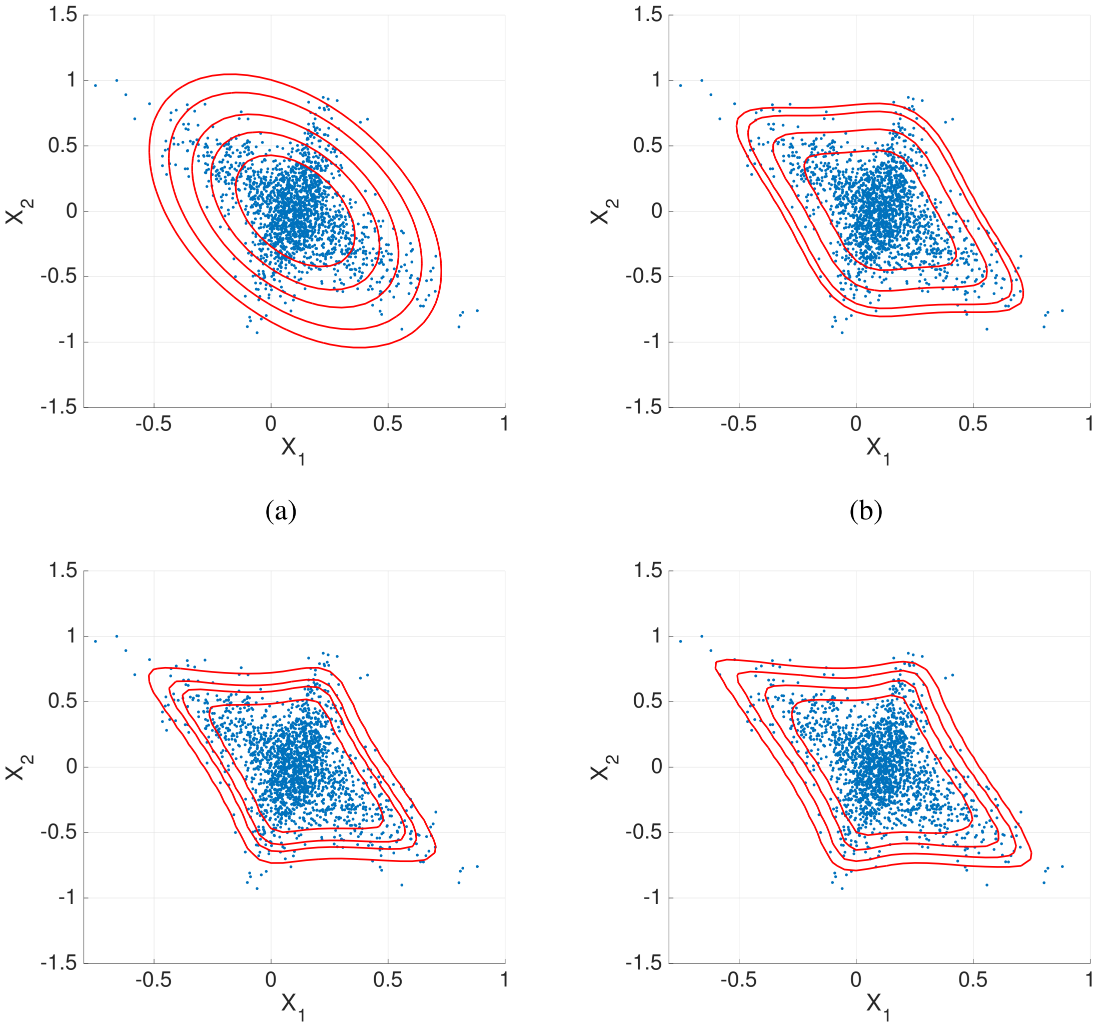


Figure 3.1: Sample images where up to second order moments are not enough to distinguish targets. Each column shows one pair of samples from VIPeR, CUHK01, PRID450s and GRID, respectively. The color bars represent the Euclidean distance between the corresponding strips within the image pair, where the left one comes from moM and the right one from GOG. The images on the first row are from the "probe" view and the second row are from the "gallery" view. In these examples, with the help of higher order moments, moM has better invariance property when the person has a fine-detailed non-Gaussian appearance, e.g., the dot-pattern in column 1, white strip on the backpack in column 2, black strip on a white shirt in column 3 and the stripe-pattern in column 4.

Furthermore, since moment matrices are on a SPD manifold, we also propose to use the on-manifold mean and flattening on its tangent space.

Experiments on five public benchmark datasets illustrate the benefits of encapsulating higher order moments information. The combination of proposed moM with GOG [10] achieves comparable or better state-of-the-art performance on all the tested datasets.



(a) (b)

(c) (d)

Figure 3.2: Level sets of (3.4) with different Ds and Ts. (a) $D = 1$; (b) $D = 2$; (c) $D = 3$; (d) $D = 4$

3.1 Empirical moment matrix

Given a dataset consisting of N samples $\mathbf{X} = \{\mathbf{x}_k\}_{k=1}^{N}$, where $\mathbf{x}_k = [x_{k1}, x_{k2}, ...x_{km}] \in \mathbb{R}^m$. The collection of all monomials of $\mathbf{x}_k \in \mathbb{R}^m$ up to order D is defined as

$$\mathbf{v}_k \in \mathbb{R}^{s_{m,D}}, \text{ with } \mathbf{v}_k(i) = x_{k1}^{d_{i1}} x_{k2}^{d_{i2}} \cdots x_{km}^{d_{im}}, \forall_{i=1}^{s_{m,D}} \tag{3.1}$$

where the tuple $\mathbf{d}_i \doteq (d_{i1}, d_{i2}, \ldots, d_{im}) \in \mathbb{N}^m$ denotes the exponents of x_{k1}, x_{k2}, ..., x_{km} in the term $\mathbf{v}_k(i)$, satisfying $0 \leq \|\mathbf{d}_i\|_1 \leq D$. The D-th[1] order empirical moment matrix is defined as

$$\begin{aligned} \mathbf{M} \doteq& \mathcal{E}\{\mathbf{v}\mathbf{v}^T\} \in \mathbb{R}^{s_{m,D} \times s_{m,D}}, \text{ with} \\ \mathbf{M}(i,j) \doteq& \mathcal{E}\{\mathbf{v}(i)\mathbf{v}(j)\} \\ =& \frac{1}{N}\sum_{k=1}^{N} \mathbf{v}_k(i)\mathbf{v}_k(j), \forall i,j = 1, \ldots, s_{m,D} \end{aligned} \tag{3.2}$$

Pauwels and Lasserre [43] show that the level set of polynomial $\mathbf{v}^T \times \mathbf{M}^{-1} \times \mathbf{v}$ can be used to represent a shape of arbitrary distribution with large enough D. When $D = 1$, the moment matrix can be written as:

$$\begin{bmatrix} 1 & \mathcal{E}(\mathbf{x}) \\ \mathcal{E}(\mathbf{x})^T & \mathcal{E}\{\mathbf{x}\mathbf{x}^T\} \end{bmatrix} \tag{3.3}$$

which is the same as the transformation from Gaussian distribution to SPD manifold, except for the normalization term [44]. Figure 3.2 demonstrates the merit of higher order moment matrix. After computing $\mathbf{M}$, we plot the level set described as (3.4) with different T by red lines in Figure 3.2:

$$\mathbf{v}^T \times \mathbf{M}^{-1} \times \mathbf{v} = T \tag{3.4}$$

As observed in the plot, moment matrices with higher D can preserve the shape of points more accurately.

Comparing to statistical properties such as mean and covariance, which are popular in the literature, moment matrix of higher order ($D \geq 2$) contains richer statistical information. Therefore, in the sequel, within a region of interest p, we will use the moment matrix $\mathbf{M}_p$ defined as (3.2) to model the local appearance feature distribution.

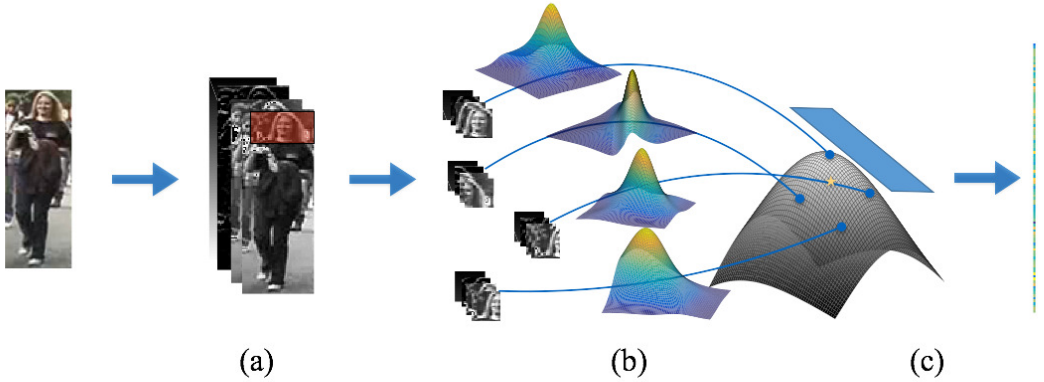


Figure 3.3: moM feature extraction: Starting with a pedestrian image, (a) pixel features are computed to extract the color and gradient information and (b) each patch is modeled with a moment matrix, which lies on a SPD manifold. (c) On-manifold mean is applied to pool the information along horizontal strips and then the mean matrix is flattened to its tangent space and vectorized to form the final descriptor of the strip.

3.2 The algorithm

Figure 3.3 shows the pipeline for extracting moM and the step-by-step procedure is shown in Alg. 1.

3.2.1 Pixel features

Following the work [10], we also use the following pixel level features to represent local appearance information:

$$\mathbf{x}_k = [y, A_0, A_{90}, A_{180}, A_{270}, C_a, C_b, C_c]^T \tag{3.5}$$

where y is the y coordinate of pixel z_k, $A_{\theta \in \{0,90,180,270\}}$ are the magnitudes of gradient along four directions, and $C_{\{a,b,c\}}$ are intensity values in the corresponding color channel. All dimensions are normalized to the range $[0, 1]$. In this paper, we will use RGB, HSV, LAB or normalized RG as the color channel.

3.2.2 On-manifold mean

As shown in Figure 3.1, pedestrians inside the bounding boxes are roughly aligned in the vertical direction. The current state-of-the-art re-id features, GOG [10] and Local Maximal

[1] Please note the D-th order $\mathbf{M}$ has moments up to order $2D$

Occurrence feature (LOMO) [42], also take advantage of this assumption and apply information pooling along the same horizontal strip. In this work, mean pooling is used to represent the patches at the same height. Since $\mathbf{M}$ is an SPD matrix, all $\mathbf{M}$s lie on an SPD manifold. Then on-manifold distance should be applied to compute the mean matrix. In this work, we adopt Log-Euclidean Riemannian Metric (LE) [45] as in (3.6) to calculate the distances between two SPD matrices:

$$\sigma_{LE}(\mathbf{M}_{p1}, \mathbf{M}_{p2}) = \| \log(\mathbf{M}_{p1}) - \log(\mathbf{M}_{p2}) \| \tag{3.6}$$

and the associated on-manifold mean for strip s is defined as follows:

$$\bar{\mathbf{M}}_s = \exp(\frac{1}{Q}\sum_{p=1}^{Q} \log(\mathbf{M}_p)) \tag{3.7}$$

where Q is the number of patches in strip s and $\exp(\cdot)$ denotes the matrix exponential operator.

The benefits of using LE as the on-manifold metric are two-fold: 1) it has a closed form solution and can be computed very efficiently; 2) to feed the feature to off-shelf metric learning methods, one can transfer the SPD matrices into Euclidean space by taking the logarithm, which will cancel the $\exp(\cdot)$.

The final vectorized moM feature $\mathbf{g}_s$ for strip s is obtained by equation (3.8)

$$\begin{aligned} \mathbf{\Gamma}_s =& \log(\bar{\mathbf{M}}_s) \\ \mathbf{g}_s =& \text{vec}(\Gamma_s) \\ =& \Big[\mathbf{\Gamma}(1,1), \sqrt{2}\mathbf{\Gamma}(1,2), \ldots, \mathbf{\Gamma}(2,2), \sqrt{2}\mathbf{\Gamma}(2,3), \ldots\Big] \end{aligned} \tag{3.8}$$

where $\log(\cdot)$ denotes the matrix logarithm operator and $\sqrt{2}$ applies on off-diagonal elements to keep the condition $\|\mathbf{\Gamma}_s\|_F = \|\mathbf{g}_s\|_2$ holding. To reduce the numerical problem caused by the logarithm of small eigenvalues of the moment matrix, all $\mathbf{M}_p$ are normalized to $\det(\mathbf{M}_p) = 1$. The final feature vector $\mathbf{f}$ is the concatenation of all $\mathbf{g}_s$ in different strips. Following the setting in [10, 9], we also apply mean removal and power normalization. Thus, the moM descriptor is normalized by (3.9)

$$\mathbf{f}_{norm} = \text{sign}(\mathbf{f} - \boldsymbol{\mu}_f)|\mathbf{f} - \boldsymbol{\mu}_f|^{0.5} \tag{3.9}$$

where $|\cdot|$ is the absolute value and $\boldsymbol{\mu}_f$ is the mean of all moM features in the training set.

Algorithm 1 moM feature extraction

```
Require: Image $I$, number of horizontal strips $S$, number of patches per strip $Q$, moment matrix order $D$,
Compute pixel features in (3.5)
for strip $s = 1$ to $S$ do
    for patch $p = 1$ to $Q$ do
        Compute moment matrix $\mathbf{M}_p$ based on (3.2)
    end for
    Compute on-manifold mean $\bar{\mathbf{M}}_s$ based on (3.7)
    Compute the feature of $\mathbf{g}_s$ based on (3.8)
end for
Concatenate $\mathbf{g}_{1,2,\ldots,S}$ to form the final moM feature $\mathbf{f}$
```

Chapter 4

Person Re-Identification

Person Re-identification (re-ID), in general, is defined as re-identifying a human of interest in a set of images or videos captured by cameras with limited or no overlapping fields of view. In the past decade, due to the emerging demands of real-world problem, such as security and video surveillance in large public area, researchers have dedicated significant amount of efforts to push the state-of-art of this problem. Figure 4.1 illustrates the divisions and sequence of modern re-ID system.

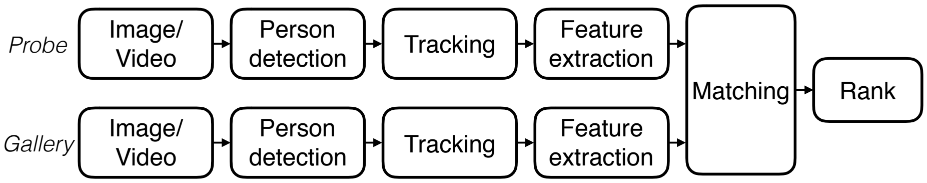


Figure 4.1: A typical end-to-end re-ID system pipeline.

Starting as part of the multi-camera tracking system, person re-ID has been separated as an independent computer vision problem by Gheissari *et al.* [46]. Later, after the release of very first re-ID dataset VIPeR [47], several single-shot methods were proposed [48, 49], which only used one shot probe image to query the correct matching in the gallery set. On the other hand, assuming the availability of multiple shots of a target person available, multi-shot re-ID also attracted the interests of many researchers [50, 51, 37]. Furthermore, by extending multiple shots to a short video clip, Wang *et al.* [52] started the work on video-based re-ID and drew a lot attentions from other researchers [53, 54, 55, 56]. Around the same time, several unsupervised re-ID methods [57, 58, 59, 60] were proposed to tackle the challenge of ground truth labeling for person re-ID. Due to the same challenge, relatively small datasets limited the adoption of deep learning methods on

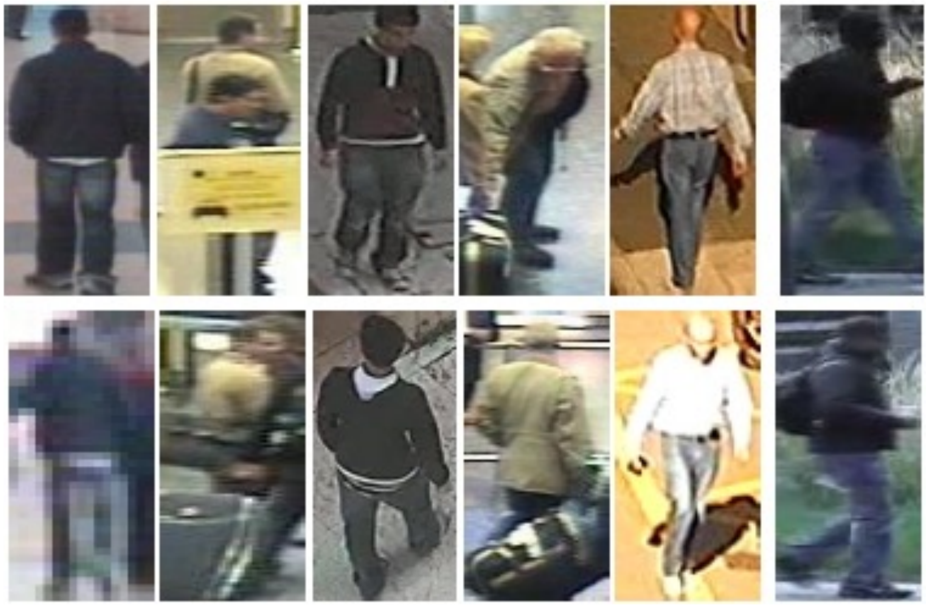


Figure 4.2: The challenge of re-ID problem

re-ID until the appearance of two large scale datasets, CUHK03 [61] and Market1501 [62]. Following that, methods based on both CNN and RNN were studied from 2014 [61, 63, 64, 65, 66, 67]. Recently, instead of assuming perfect Bounding boxes (BBox) and trajectories from manual labeling, several end-to-end re-ID datasets with automatically person detector and tracking modules were released to test the robustness of new methods. For more details, we refer the reader to [68, 69, 70, 71].

Even though the re-ID system can obtain perfect BBox from the person detector and tracker, it is still a very challenging problem because of the severe intra-class variance between different cameras. Figure 4.2 shows the challenge of a typical re-ID problem. The example of hard cases are, from left to right, resolution, occlusion, viewpoints, different poses, illumination and similar appearance of different people.

Most of the existing re-id literature focuses on two aspects of the problem: 1) designing viewpoint invariant feature descriptors [37, 40, 38, 42, 39, 72, 41, 10, 73, 74] and/or 2) learning a supervised classifier to alleviate the effect of the variances across the cameras [75, 42, 36, 76, 77, 48, 78, 79, 80, 81]. Recently, deep neural networks have been adopted to learn both the descriptor and classifier simultaneously [63, 82, 61, 64]. For more details, we refer the reader to [68, 69, 70, 71].

Person re-id specific hand-crafted features mainly focus on the invariance across different cameras. In [50], based on the symmetric axis of each body part, a carefully designed body configuration was modeled. Then, the weighted color histogram was computed, depending on the distance between the pixel and the axis. The final representation was also combined with maximally stable color regions (MSCR) and recurrent high-structured patches (RHSP). Ma *et al.* [72] used the biological inspired feature (BIF) as the raw feature and compressed it using the similarity between the covariance matrices of small patches. Since then, following the development of metric learning methods, researchers tend to use native but redundant features to feed into the supervised learned metric. Gary and Tao [47] used 8 color channels (RGB, HS, and YUV) and 21 texture filters. In [83], responses of texture filters were substituted by LBP features. Instead of color histogram, in [84], the local mean of each patch was adopted. Pedagadi *et al.* [77] added the first three moments to the color histogram to represent a small patch. More recently, Zhao *et al.* [74] combined the LAB histogram with dense SIFT descriptors on a densely sampled grid. To obtain a stable representation, color names have been applied recently. In [73], salience color name distributions were computed over different color models to remedy the illumination variance. Zheng *et al.* [62] encoded the local color name descriptors through Bag-of-Words. Liao *et al.* [42] proposed maximum-pooling the color and SILTP [85] histogram along the same horizontal strip to achieve better viewpoint invariance.

Covariance and Gaussian descriptors have been applied in person re-id, to compress more information than histogram and local mean. In [37], pixel level color intensity and gradient in a local patch are compressed into a covariance matrix. Ma *et al.* [39] modeled the low level feature with a Gaussian distribution and compare the Gaussian with the product on Lie group. In [41], GMM is used to model the pixel feature by assuming the variants are independent of each other. Inspired by LOMO [42], a hierarchical Gaussian feature (GOG) was proposed in [10]. Similar to previous work, pixel features in a small patch are modeled by a Gaussian distribution, which is embedded in an SPD manifold. Then, the second level models the distribution of the first level descriptors within a strip around the same height.

4.1 Datasets

We evaluate the proposed moM feature using four widely used hand labeled re-id benchmark datasets and one large-scale automatically detected dataset.

VIPeR [47] contains images of 632 persons. Each person has two images taken from different viewpoints. All identities are separated into training and testing sets equally. One view is fixed as the probe view. This procedure was repeated 10 times and the average performance is reported.

QMUL underGround Re-IDentification (GRID) [86] dataset has 250 paired pedestrians and 775 un-paired distractions captured in a subway station. The large size of the gallery set and relatively low image quality make it one of the most challenging re-id datasets. We use the provided partition configuration.

PRID450s [87] is a subset of PRID2011 [38] with 450 persons and 2 cameras. Each person has one image per camera. Similar to VIPeR dataset, the train and test sets are equally separated and one camera is fixed as the probe one. Ten repeated evaluations are performed and the average result is reported.

Market1501 [62] dataset is a recently proposed large scale re-id dataset. It contains 1501 identities from 6 cameras. All bounding boxes are automatically detected with DPM [50] algorithm and manually annotated. In total, it contains 32,668 bounding boxes including 2,793 false alarms from the person detector. We adopt the provided train/test partition to evaluate our feature.

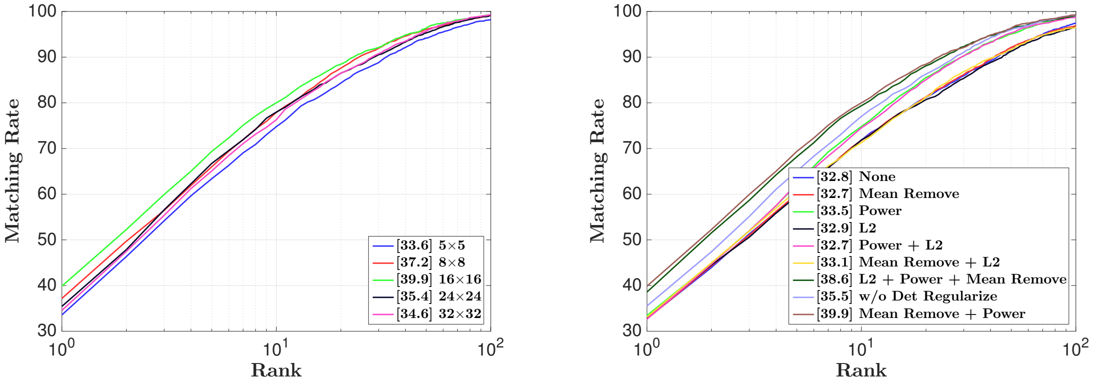
4.2 Implementation details

First of all, we reshape all images to the size of 128×48, and the patch size is set to 16×16 with 50% overlap, which will generate 15 horizontal strips. The order D is set to 2, so the moment matrix size is 45×45. Therefore, the final dimension of the feature with RGB is $(45 \times 46/2) \times 15 = 15,525$. Following the setting in [10], we weight the patches according to their position on x axis as $w_p = e^{-(x_p - x_c)^2/2\sigma^2}$, where $x_c = W/2$ and $\sigma = W/4$. x_p is the x coordinate of the center point of patch p and W is the width of the image. We also fuse moM from different color channels to boost the performance. Results with fused feature are noted with subscripts “f”. For all the experiments, kLFDA with linear kernel [78] is used as the metric learning algorithm.

4.3 Experiments

4.3.1 Method analysis

Patch size: Figure 4.3(a) shows the results with different patch sizes. For a fair comparison, we keep the adjacent patches with 50% overlapping. We note that the performance decreases when the patch size is either too small or too large. On the one hand, there is not enough number of pixels within small patches to estimate the higher order moment matrix. Moreover, small patches tend to be less discriminant because they only model local information. As shown in the results, the rank 1



(a) (b)

Figure 4.3: Performance analysis on VIPeR dataset.

performance downgrades 6.3% when the patch size shrinks to 5×5. On the other hand, although large patches provide enough samples to estimate complex distributions, they encode specific pose and lose multi-view invariance. Therefore, we choose a median patch size of 16×16 to compromise the discriminating and invariant properties.

Normalization: Figure 4.3(b) illustrates the effects of applying different normalizations. By forcing the product of eigenvalues equal to 1, the determinant normalization improves the result by 4.4%. Because most of the elements of higher order moments are small numbers, their logarithms are large negative values, which overwhelm the variance on that dimension. The mean removal step will center all dimensions while keeping the variance at the same time. The power normalization reduces the "spike" situation further more. Combining these two steps improves the rank 1 accuracy by 7.1%. Adding ℓ_2 normalization decreases it by 1.3%.

Moment matrix order: In Table 4.1, the first three rows show the results with $D = 1$. Comparing to the results with the following three rows with $D = 2$, the average rank1 performance with different on-manifold means increases by 7.4%, 9.8%, 16.3% and 5.5% on VIPeR, CUHK01, PRID450s and GRID, respectively. One can also observe a distinct margin between the blue curves and the other curves in Figure 4.4. This shows that the higher order moments are informative to boost the performance of the descriptor.

On-manifold metric: Besides the LE metric, there are several other metrics for the SPD manifold. Here, we also compare the performance using Jeffery Divergence (Jeff) [88] and Jensen-Bregman Log-det Divergence (JBLD) [89] on four datasets. Experimental results are shown in

Table 4.1 with different superscripts. These means are defined as:

$$\bar{\mathbf{M}}_{JBLD}^{(t+1)} = \left[\frac{1}{Q} \sum_{p=1}^{Q} \left(\frac{\mathbf{M}_p + \bar{\mathbf{M}}_{JBLD}^{(t)}}{2} \right)^{-1} \right]^{-1} \tag{4.1}$$

$$\begin{gathered} \bar{\mathbf{M}}_{Jeff} = \mathbf{P}^{-1/2}(\mathbf{P}^{1/2}\mathbf{Q}\mathbf{P}^{1/2})^{1/2}\mathbf{P}^{-1/2}, \text{ with} \\ \mathbf{P} = \sum_{p=1}^{Q} \mathbf{M}_p^{-1}, \mathbf{Q} = \sum_{p=1}^{Q} \mathbf{M}_p. \end{gathered} \tag{4.2}$$

When only using the RGB as the color channels, although moM$_{rgb}^{Jeff}$ outperforms moM$_{rgb}^{LE}$ by 0.9% on VIPeR rank1 result, moM$_{rgb}^{LE}$ beats the others on CUHK01, PRID450s and GRID datasets. On average, moM$_{rgb}^{LE}$ achieves 2.3% and 1.3% higher rank1 performance along the four datasets compared with moM$_{rgb}^{Jeff}$ and moM$_{rgb}^{JBLD}$, respectively. When fusing with other color channels and the GOG feature, moM with LE performs slightly better than JBLD.

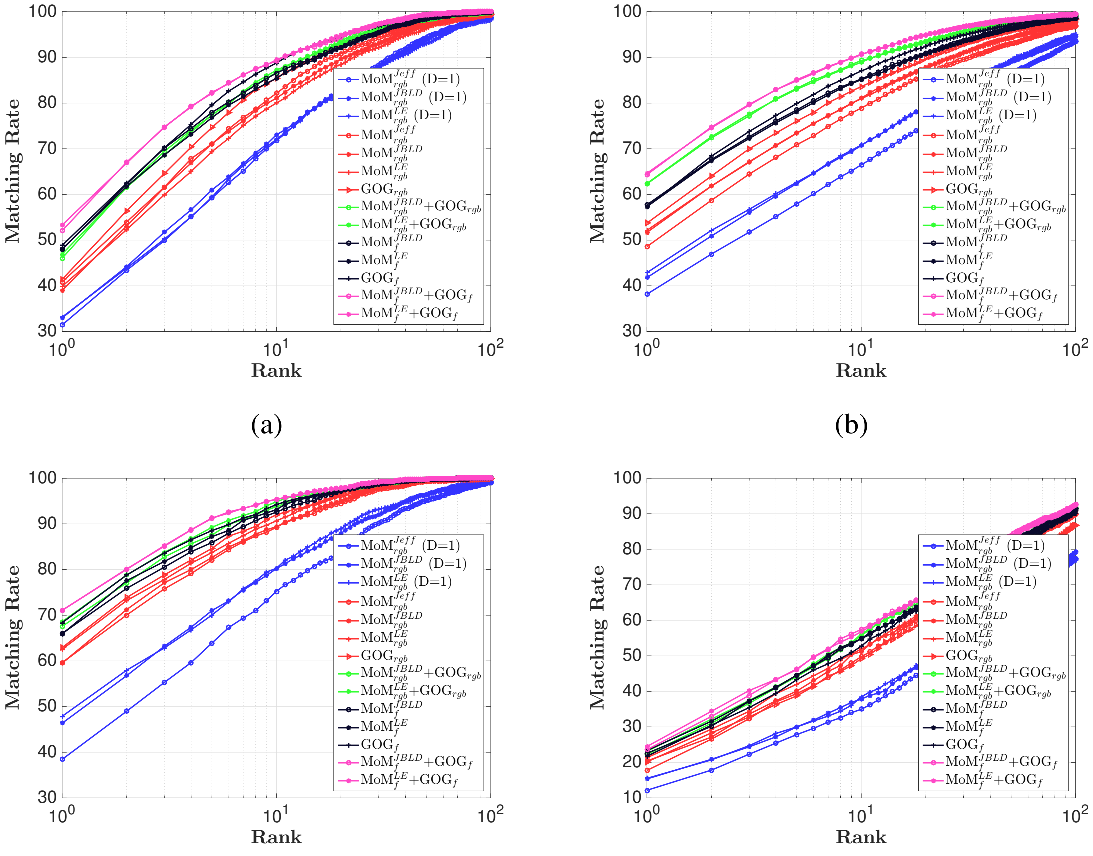
Table 4.1: Comparing with different Ds and on-manifold means. The best results in each dataset are marked in red.

Dataset	VIPeR				CUHK01				PRID450s				GRID			
Methods	r=1	r=5	r=10	r=20	r=1	r=5	r=10	r=20	r=1	r=5	r=10	r=20	r=1	r=5	r=10	r=20
$\mathrm{moM_{rgb}}^{\mathrm{Jeff}}$ (D=1)	31.4	59.2	71.8	82.7	38.2	57.9	66.5	75.2	38.4	63.9	75.2	84.6	12.2	27.8	35.0	45.8
$\mathrm{moM_{rgb}}^{\mathrm{JBLD}}$ (D=1)	33.0	61.0	73.1	82.8	41.8	62.4	70.7	79.2	46.4	71.0	80.2	87.9	15.4	30.0	37.8	48.5
$\mathrm{moM_{rgb}}^{\mathrm{LE}}$ (D=1)	33.1	59.4	72.1	82.7	42.9	62.7	70.8	79.3	47.9	70.0	80.3	88.9	15.5	30.0	38.5	48.6
$\mathrm{moM_{rgb}}^{\mathrm{Jeff}}$	40.8	71.0	82.1	90.8	48.6	70.9	78.9	86.4	59.6	82.1	89.2	94.5	17.8	39.4	49.8	62.2
$\mathrm{moM_{rgb}}^{\mathrm{JBLD}}$	39.0	71.1	81.1	89.6	51.7	73.4	81.1	87.8	59.5	82.6	89.4	95.0	20.4	40.1	51.2	62.7
$\mathrm{moM_{rgb}}^{\mathrm{LE}}$	39.9	69.4	80.0	88.4	52.1	73.4	80.9	87.6	62.5	83.7	90.7	96.5	21.4	42.0	51.9	62.6
$\mathrm{GOG_{rgb}}$	41.4	74.7	85.4	92.6	53.7	76.0	83.6	89.8	62.9	84.6	92.0	96.1	20.2	38.7	49.2	59.8
$\mathrm{moM_{rgb}}^{\mathrm{JBLD}}$+$\mathrm{GOG_{rgb}}$	46.0	77.3	86.7	93.8	62.3	83.2	89.3	93.5	67.6	87.6	93.8	97.4	22.2	44.2	55.7	66.1
$\mathrm{moM_{rgb}}^{\mathrm{LE}}$+$\mathrm{GOG_{rgb}}$	46.9	77.4	87.2	93.0	62.4	83.0	88.9	93.3	68.6	89.2	94.8	97.4	23.1	44.5	56.2	66.7
$\mathrm{moM_f}^{\mathrm{JBLD}}$	48.0	77.9	86.6	92.2	57.7	78.5	85.3	90.8	66.0	85.9	92.6	97.1	22.6	44.6	54.9	64.8
$\mathrm{moM_f}^{\mathrm{LE}}$	48.0	76.8	85.4	92.1	57.3	78.1	85.1	90.7	65.9	87.2	93.1	97.2	23.4	44.6	54.8	65.4
$\mathrm{GOG_f}$	48.8	79.6	88.8	94.6	57.3	79.9	87.0	92.5	68.4	88.5	94.2	97.2	21.8	43.3	52.7	63.5
$\mathrm{moM_f}^{\mathrm{JBLD}}$+$\mathrm{GOG_f}$	52.1	82.1	89.2	94.5	64.3	**85.1**	**90.7**	**94.9**	**71.1**	91.2	**95.4**	97.8	23.6	**46.3**	**57.4**	**67.4**
$\mathrm{moM_f}^{\mathrm{LE}}$+$\mathrm{GOG_f}$	**53.3**	**82.3**	**89.5**	**94.8**	**64.6**	84.9	90.6	94.8	**71.1**	**91.3**	**95.4**	**97.9**	**24.5**	46.1	56.8	66.9

4.3.2 Comparison with GOG descriptor

The results in Table 4.1 and Figure 4.4 compare the performances of the moM features, the GOG features and the combination of both of them. We ran the code provided by the authors of [10] and set the patch size to 15×15[1] and number of strips to 15. Among all four datasets, $\mathrm{moM_{rgb}}^{\mathrm{LE}}$

[1] The code provided can only accept odd number as the patch size



(a)

(b)

(c)

(d)

Figure 4.4: CMC curves for (a)VIPeR, (b)CUHK01, (c)PRID450s and (d)GRID datasets

obtains slightly worse results in VIPeR and CUHK01, comparable result in PRID450s and better result in GRID. When fusing with all different color channels, $\mathrm{moM_f}^{\mathrm{LE}}$ performs worse in VIPeR and PRID450s, comparable in CUHK01 and better in GRID. However, by simply concatenating moM and GOG, a consistent out-performance can be achieved. In Figure 4.4, one can observe a clear margin between green and red curves and pink and black curves. Specifically, with the RGB color channel, adding $\mathrm{moM_{rgb}}^{\mathrm{LE}}$ to $\mathrm{GOG_{rgb}}$ improves the rank 1 performance by 5.5%, 8.7%, 5.7% and 2.9%, respectively. After fusing with all different color channels, adding $\mathrm{moM_{rgb}}^{\mathrm{LE}}$ to $\mathrm{GOG_{rgb}}$ further improve the rank 1 performance by 4.5%, 7.3%, 2.7% and 2.7%, respectively. The result implies that moM and GOG features encapsulate complementary appearance informations. In the following, we will name this combination ac Combination of Mean of Moment matrix feature and Gaussian of Gaussian feature (moMaGO).

This observation can be explained by noting that the GOG feature has up to 2nd order information of the distribution of the distributions representing the patches. However, it contains no information about the higher order (greater than 2) moments of these patches. On the other hand, the moM feature has information about the mean value (across patches) of the higher order moments, but not about their variance. Thus, one can think of the combination of GOG and moM as a tractable approximation to a "Moments of Moments" feature, where GOG provides information about the variance of 1st and 2nd order moments while moM provides information about the mean value of all moments (up to 4th order). For $\mathbf{x} \in \mathbb{R}^8$ this leads to a feature vector with $O(10^3)$ elements, as opposed to a true Moment of Moments feature ($D = 2$) that would have $O(10^6)$ elements.

Figure 4.5 gives two qualitative analysis examples. When the identity has fine-detailed appearance patterns, moM feature preserves those patterns better than the GOG feature. In the first example, moM feature captures the backpack with rich texture in the probe image and retrieves the gallery images with similar pattern to top two and finds the correct matching at rank 1. On the other hand, when the identity has homogeneous local texture but relatively complex patterns along the strip, GOG feature is preferred. In the second example, the strip level second order moment helps to preserve the blue/black/skin color pattern along the upper body part.

4.3.3 Comparison with state-of-the-art methods

In Table 4.2, we compare the combination of $\mathrm{moM_f}^{\mathrm{LE}}$ and $\mathrm{GOG_f}$[2] with recently published re-id methods. In four datasets, we achieve new state-of-the-art performance on CUHK01 and

[2] Please note that the GOG feature we used has a different setting from [10]

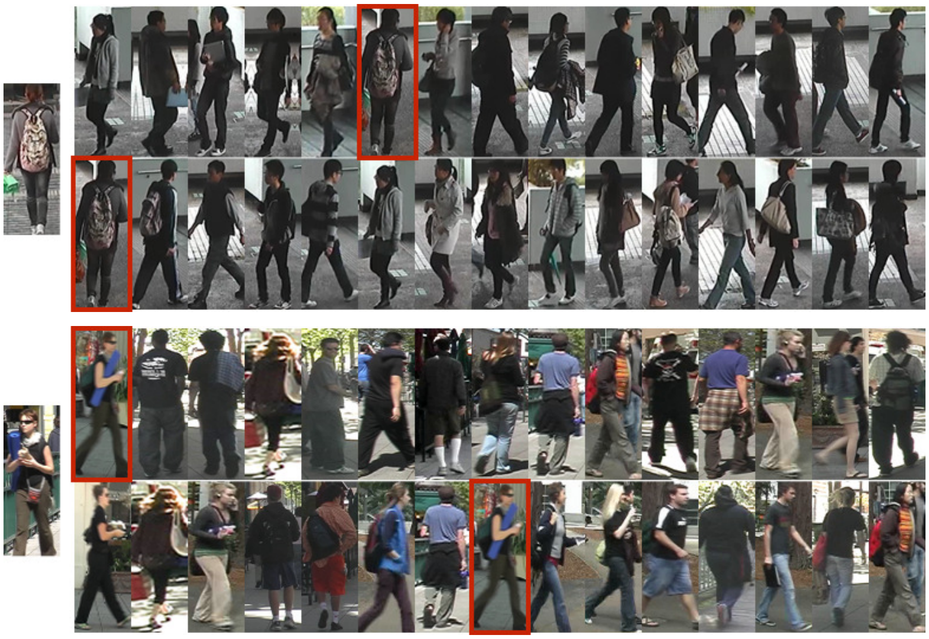


Figure 4.5: Examples for moM and GOG features. The very left image is the probe image and the first row on the right hand side is the result from GOG_{rgb} and the second row is from $\text{moM}_{\text{rgb}}{}^{\text{LE}}$. The correct match is labeled with a red box. The first example shows the situation moM feature is preferred while the second one shows the case GOG feature is better. Please see the text for more analysis.

Table 4.2: Comparison with state-of-the-art methods. The best results in each dataset are marked in red and the second best in blue.

		VIPeR				CUHK01				PRID450s				GRID			
Methods	Reference	r=1	r=5	r=10	r=20	r=1	r=5	r=10	r=20	r=1	r=5	r=10	r=20	r=1	r=5	r=10	r=20
moMaGO^2	**Ours**	**53.3**	**82.3**	**89.5**	94.8	**64.6**	**84.9**	**90.6**	**94.8**	**71.1**	**91.3**	**95.4**	**97.9**	**24.5**	**46.1**	**56.8**	**66.9**
SSDAL + XQDA	ECCV16[66]	43.5	71.8	81.5	89.0	-	-	-	-	-	-	-	-	22.4	39.2	48.0	58.4
SCSP	CVPR16[90]	**53.5**	**82.6**	**91.5**	**96.7**	-	-	-	-	-	-	-	-	24.2	44.6	54.1	65.2
GOG_f + XQDA	CVPR16[10]	49.7	79.7	88.7	94.5	**57.9**	**79.2**	86.2	92.1	**68.0**	**88.7**	**94.4**	**97.6**	**24.8**	**47.0**	**58.4**	**68.9**
TCP	CVPR16[82]	47.8	74.7	84.8	91.1	53.7	84.3	**91.0**	**96.3**	-	-	-	-	-	-	-	-
SS-SVM	CVPR16[80]	42.7	-	84.3	91.9	-	-	-	-	60.5	-	88.6	93.6	22.4	-	51.3	61.2
MLAPG	ICCV15[36]	40.7	-	82.3	92.4	-	-	-	-	-	-	-	-	16.6	-	41.2	53.0
Metric Ensumble	CVPR15[76]	45.9	77.5	88.9	**95.8**	53.4	76.4	84.4	90.5	-	-	-	-	-	-	-	-
LOMO+XQDA	CVPR15[42]	40.0	-	80.5	91.1	49.2	75.5	84.2	90.8	62.6	85.6	92.0	96.6	16.6	-	41.8	52.4
SCNCD	ECCV14[73]	37.8	68.5	81.2	90.4	-	-	-	-	41.6	68.9	79.4	87.8	-	-	-	-

Table 4.3: Comparison with state-of-the-art on Market1501 dataset.

Method	Reference	r=1	mAP
moM$_f^{LE}$+GOG$_f$	**Ours**	**71.6**	**43.5**
moM$_f^{LE}$	**Ours**	61.0	30.3
GOG$_f$	CVPR16 [10] [2]	**66.7**	38.5
Gated S-CNN	ECCV16[64]	65.9	**39.6**
S-LST	ECCV16[65]	61.6	35.3
SSDAL+XQDA	ECCV16[66]	39.4	19.6
SCSP	CVPR16[90]	51.9	26.4
DNS	CVPR16[79]	55.4	29.9
BoW+KISSME	ICCV15[62]	44.4	20.8

PRID450s and outperform the second best by 6.7% and 3.1%, respectively. In VIPeR and GRID it attains the second best performance with only slightly loss. To show the generalization of moM on a large scale, automatically detected, dataset we compare with state-of-the-art works on the Market1501 dataset in Table 4.3. To be consistent with previous experiments, we report the result of GOG$_f$ with the same setting in Table 4.1. By combing with our proposed moM feature, the complementary information brings a 4.9% improvement on rank 1 performance and increases by 5% on mAP, setting a new state-of-the-art in this dataset.

Chapter 5

New Datasets and Benchmark for Person Re-Identification

Existing re-id algorithms are typically evaluated on datasets that are either hand-curated or pruned with a person detector to contain sets of bounding boxes for the probes and the corresponding matching candidates. On the other hand, real-world end-to-end surveillance systems include automatic detection and tracking modules, as depicted in Figure 4.1, that generate candidates on-the-fly, resulting in gallery sets that are dynamic in nature. Furthermore, errors in these modules may result in bounding boxes that may not accurately represent a human [91]. As noted in our recent benchmark paper [92], the size of a dataset, in terms of both number of identities as well as number of bounding boxes, is critical to achieve good performance. Furthermore, in real-world end-to-end surveillance systems, as noted in Camps *et al.* [91], we can use camera calibration information to predict motion patterns, potentially helping to prune out irrelevant candidates and reducing the search space. While these issues are critical in practical re-id applications, they are not well-represented in the currently available datasets. To this end, we propose two **new, large-scale datasets** constructed from images captured in a challenging surveillance camera network.

We also present an up-to-date performance benchmark for these datasets, in which we test 10, including the proposed moM and moMaGO, different features and 12 different metric learning methods. The goal is to validate the effectiveness of statistical moment modeling feature and systematically study how existing re-ID algorithms fare on the new datasets. The code library has been published and can be accessed via https://github.com/RSL-NEU/person-reid-benchmark.

5.1 re-ID Datasets

Table 5.1: An overview of existing widely used re-id datasets.

Dataset	Year	# people	# BBox	# FP	# distractors	# cameras	Environment	Label source	Video?	Full frame?
VIPeR [47]	2007	632	1,264	0	0	2	-	hand	N	N
ETHZ [93]	2007	148	8,580	0	0	1	-	hand	N	N
QMUL iLIDS [94]	2009	119	476	0	0	2	airport	hand	N	N
GRID [95]	2009	1,025	1,275	0	775	8	subway	hand	N	N
3DPeS [96]	2011	192	1,011	0	0	8	campus	hand	N	Y
PRID2011 [38]	2011	934	24,541	0	732	2	campus	hand	Y	Y
CAVIAR4ReID [51]	2011	72	1,220	0	22	2	mall	hand	N	Y
V47 [97]	2011	47	752	0	0	2	-	hand	N	Y
WARD [98]	2012	70	4,786	0	0	3	-	hand	Y	N
SAIVT-Softbio [99]	2012	152	64,472	0	0	8	campus	hand	Y	Y
CUHK01 [100]	2012	971	3,884	0	0	2	campus	hand	N	N
CUHK02 [101]	2013	1,816	7,264	0	0	10 (5 pairs)	campus	hand	N	N
CUHK03 [61]	2014	1,467	13,164	0	0	10 (5 pairs)	campus	hand/DPM [102]	N	N
HDA+ [103]	2014	53	2,976	2,062	20	13	office	hand/ACF [104]	N	Y
RAiD [105]	2014	43	6,920	0	0	4	campus	hand	N	N
iLIDS-VID [52]	2014	300	42,495	0	0	2	airport	hand	Y	N
Market1501 [62]	2015	1,501	32,217	2,798+500K	0	6	campus	hand/DPM [102]	N	N
MARS [56]	2016	1,261	1,191,003	147,744	0	6	campus	DPM [102]+GMMCP [106]	Y	N
DukeMTMC-reID [107]	2017	1,812	36,441	0	408	8	campus	hand	N	Y
DukeMTMC4ReID	**2017**	**1,852**	**46,261**	**21,551**	**439**	**8**	**campus**	**Doppia** [108]	**N**	**Y**
Airport	**2017**	**9,651**	**39,902**	**9,659**	**8,269**	**6**	**airport**	**ACF [104]**	**N**	**N**

Table 5.1 provides a statistical summary of these datasets. In the table and following content, we define an identity as a person with images in both the probe and gallery cameras, a distractor as a person only appearing in one camera, and an FP as a false alarm from the person detector.

VIPeR [47] is one of the earliest available and most widely used datasets, consisting of 632 identities from two disjoint camera views. GRID [95] has 250 paired identities across 8 cameras, in addition to 775 distractor identities to mimic a realistic scenario. 3DPeS [96] consists of 1,011 images corresponding to 192 identities, captured in an 8-camera network. PRID2011 [38] is constructed in an outdoor environment, with 200 paired identities captured in two camera views. CAVIAR4ReID [51] is constructed from two cameras placed inside a shopping mall, with 50 paired identities available. V47 [97] captures 47 identities in an indoor environment. WARD [98] captures 70 identities in a 3-camera network. SAIVT-Softbio [99] captures 152 identities in an 8-camera surveillance network installed on a campus. HDA+ [103] captures 53 identities in an indoor environment, in addition to a number of distractor identities for the gallery. RAiD [105] captures 43 identities as seen from two indoor and two outdoor cameras. iLIDS-VID [52] captures 300 identities in an indoor surveillance camera network installed in an airport. Market1501 [62] captures 1,501 identities in addition to

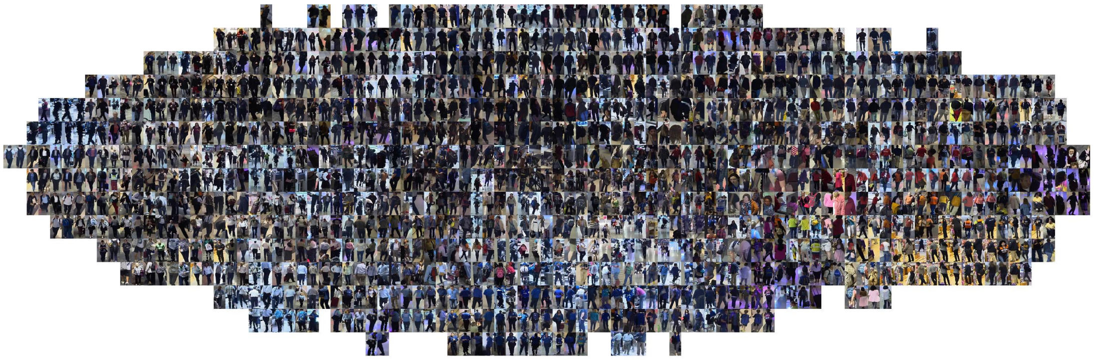


Figure 5.1: Samples of images from the proposed Airport dataset.

2,798 false positives and 500k distractors, providing for a realistic gallery. Airport [92] represents a realistic scenario in which 1,382 identities are captured in a 6-camera indoor surveillance network in an airport. All images are automatically generated by means of an end-to-end re-id system [91, 109]. MARS [56] is a video extension of the Market1501 dataset, with long-duration image sequences captured for 1,261 identities.

5.1.1 Airport Dataset

The airport dataset was collected using a surveillance network with six cameras. It covers a secure area within a mid-sized airport with one checkpoint and three possible connected concourses. All cameras recorded 12-hour long videos from 8 AM to 8PM with 768 × 432 resolution at 30 frames per second. Because of the restricted covered area, we assume the target person spends limited time inside the camera network. Therefore, we randomly picked 40 five minutes long video clips for each long video. To mimic the real world application, bounding boxes of the person are generated with a real-time end-to-end prototype described in [91]. Specifically, Aggregated Channel Features (ACF) [104] is adopted to generate person detections and the combination of FAST corner features [19] and the KLT tracker [110] is utilized to track people and associate detections. The dataset can be requested at http://www.northeastern.edu/alert/transitioning-technology/alert-datasets/alert-airport-re-identification-dataset/.

Unlike other datasets that capture image data from public environments such as universities [111, 112] [56] [113], shopping locations [51], or publicly accessible spots in transportation gateways [114], the Airport dataset provides data captured from video streams inside the secure area, post the security checkpoint, of a major airport. It is generally very difficult to obtain data from such a camera network, in which configuration settings (e.g., network topology and placement of cameras) are driven by security requirements. For instance, most academic datasets have images taken from cameras with optical axes parallel to the ground plane, as opposed to the real world where the angle is usually much larger due to constraints on where and how the cameras can be installed. This aspect is explicitly captured by the Airport dataset. Unlike other datasets that primarily capture images of people in a university setup (e.g., Market1501, CUHK) , the Airport dataset captures images of people from an eclectic mix of professions, leading to a richer, more diversified set of images. Another key difference with existing datasets is the temporal aspect; we capture richer time-varying crowd dynamics, i.e., the density of people appearing in the source videos naturally varies according to the flight schedule at each hour. Such time-varying behavior can help evaluate the temporal performance of re-id algorithms, an understudied area [115].

Since all the bounding boxes were generated automatically without any manual annotation, this dataset accurately mimics a real-world re-id problem setting. A typical fully automatic re-id system should be able to automatically detect, track, and match people seen in the gallery camera, and the proposed dataset exactly reflects this setup. In total, from all the short video clips, tracks corresponding to 9,651 unique people were extracted. The number of bounding box images in the dataset is 39,902, giving an average of 3.13 images per person. The sizes of detected bounding boxes range from 130×54 to 403×166. 1,382 of the 9,651 people are paired in at least two cameras. A number of unpaired people are also included in the dataset to simulate how a real-world re-id system would work: given a person of interest in the probe camera, a real system would automatically detect and track all the people seen in the gallery camera. Therefore, having a dataset with a large number of unpaired people greatly facilitates algorithmic re-id research by closely simulating a real-world environment. While this aspect is discussed in more detail in our system paper [116], we briefly describe how this dataset can be used to validate detection and tracking algorithms typically used in an end-to-end re-id system. Specifically, since we have both valid and invalid detections in our dataset, we can use them interchangeably to evaluate the impact of the detection module. For instance, adding invalid detections to the gallery would help evaluate the need for more detection accuracy at the cost of computation time. Since we have access to multiple broken tracklets for each person, we can interchangeably use them to evaluate the impact of the tracking module. For instance, manually associating all broken tracklets can help evaluate the need for more tracking accuracy at the cost of computation time. We can also fuse these two concepts together to evaluate the need for more detection and tracking accuracy together, helping understand the upper-bound performance of real-world systems. A sample of the images available in the dataset is shown in Figure 5.1. As

Figure 5.2: Samples of images from the proposed DukeMTMC4ReID dataset

can be seen from the figure, these are the kind of images one would expect from a fully automated system with detection and tracking modules working in a real-world surveillance environment.

5.1.2 DukeMTMC4ReID

The DukeMTMC4ReID dataset was derived from the DukeMTMC dataset for multi-target tracking [112]. We note that Zheng *et al.* [107] also recently proposed a re-id dataset, called DukeMTMC-reID, based on DukeMTMC. However, our proposed dataset is significantly different on several fronts. While DukeMTMC-reID uses manually labeled ground truth, the proposed dataset uses person detections from an automatic person detector. Furthermore, DukeMTMC-reID does not include any false alarms from the detector in the gallery, while the proposed dataset has over 20,000 false alarms. Therefore, the proposed dataset is more realistic in the sense that it mimics how a practical re-id system would work in the real world.

Table 5.2: Basic statistics of the proposed DukeMTMC4ReID dataset

	Total	cam1	cam2	cam3	cam4	cam5	cam6	cam7	cam8
# bboxes	46,261	10,048	4,469	5,117	2,040	2,400	10,632	4,335	7,220
# person bboxes	24,710	4,220	4,030	1,975	1,640	2,195	3,635	2,285	4,730
# “FP” bboxes	21,551	5,828	439	3,142	400	205	6,997	2,050	2,490
# persons	1,852	844	806	395	328	439	727	457	946
# valid ids	1,413	828	778	394	322	439	718	457	567
# distractors	439	16	28	1	6	0	9	0	379
# probe ids	706	403	373	200	168	209	358	243	284

All frames in the DukeMTMC dataset were captured by 8 static cameras on the Duke University campus in 1080p and at 60 frames per second (Figure 5.3). In total, more than 2,700 people were labeled with unique IDs in eight 75-minute videos. The tight bounding boxes of each person for each frame are generated based on background subtraction and manually labeled foot positions in a few frames. Regions of interest (normal paths on the ground plane) and calibration data are also provided. The entire dataset is split into three parts: one training/validation set labeled “trainval” and two testing sets labeled “test-hard” and “test-easy”. To date, only labels from the “trainval” set have been released, which contains 1,852 unique identities in eight 50-minute videos (dataset frames 49,700–227,540).

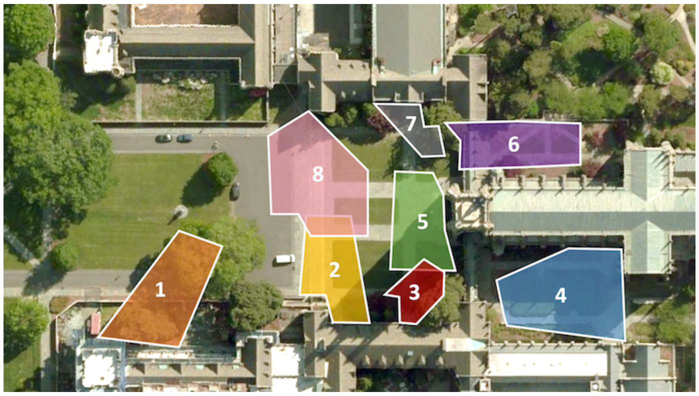


Figure 5.3: Layout of the cameras in the DukeMTMC dataset (from [1])

Based on this dataset, we constructed a large-scale real-world person re-id dataset: **DukeMTMC4ReID**. Following the recently proposed Market1501 [62] and CUHK03 [61] datasets, bounding boxes from an off-the-shelf person detector are used to mimic real-world systems. We used a fast state-of-the-art person detector [108] for accurate detections, which are filtered using predefined regions of interest to remove false alarms, e.g., bounding boxes on walls or in the sky. Then, following Market1501, based on the overlap ratio between the detection and ground truth (i.e., the ratio of the intersection to the union), we label the bounding box as “good” if the ratio is greater than 50%, false positive (“FP”) if the ratio is smaller than 20%, and “junk” otherwise. For each identity, we uniformly sample 5 “good” bounding boxes in each available camera, while retaining all the “FP” bounding boxes in the corresponding frames. To summarize, the relevant statistics of the proposed DukeMTMC4ReID dataset are provided below:

- Images corresponding to 1,852 people existing across all the 8 cameras

- 1,413 unique identities with 22,515 bounding boxes that appear in more than one camera (valid identities)
- 439 distractor identities with 2,195 bounding boxes that appear in only one camera, in addition to 21,551 "FP" bounding boxes from the person detector
- The size of the bounding box varies from 72×34 pixels to 415×188 pixels

Table 5.2 tabulates these and other statistics of the proposed DukeMTMC4ReID dataset. The dataset can be downloaded at https://github.com/NEU-Gou/DukeReID.

5.2 Benchmark

Next, we present the details of our systematic experimental evaluation of 10 existing feature extraction algorithms and 12 existing metric learning algorithms for re-id, producing an up-to-date benchmark on the proposed dataset.

5.2.1 Feature Extraction

Following the protocol described in [92], we evaluated 7 different feature extraction algorithms published up through CVPR 2016 (Table 5.3), which we briefly describe next. ELF [47] extracts color features from the RGB, YCbCr and HS color channels and texture features from the responses of multiple Schmid and Gabor filters. In HistLBP, Xiong *et al.* [78] substituted the Schmid and Gabor texture responses with LBP features, while Dense Color SIFT feature (SDC) [57] uses dense SIFT features. gBiCov [72] uses the covariance descriptor to encode multi-scale biological-inspired features. Local Descriptors encoded by Fisher Vector (LDFV) [41] uses the Fisher vector representation to encode local pixel-level information. Weighted Histograms of Overlapping Stripes feature (WHOS) [81] extract the color histogram and LBP and weighed with a Gaussian mask to remove the background. LOMO [42] extracts HSV color histogram and scale-invariant LBP features from the image in conjunction with multi-scale retinex preprocessing.

5.2.2 Metric Learning

Table 5.4 lists all the metric learning methods that were evaluated, which we briefly describe next. Fisher discriminant analysis (FDA) [118], Local Fisher Discriminant Analysis (LFDA) [77], Marginal Fisher Analysis (MFA) [119], and cross-view quadratic discriminant analysis (XQDA)

Table 5.3: Evaluated features

Feature	Source
ELF [47]	ECCV 08
LDFV [41]	ECCVW 12
gBiCov [72]	BMVC 12
SDC [57]	CVPR 13
HistLBP [78]	ECCV 14
LOMO [42]	CVPR 15
WHOS [81]	TPAMI 15
GOG [10]	CVPR 16
moM [117]	CVPRW 17
moMaGO [117]	CVPRW 17

[42] all solve eigenvalue problems based on general discriminant analysis to learn the distance metric. Xiong *et al.* [78] proposed kernelized variants of LFDA and MFA. Discriminative Null Space Learning (NFST) [79] force the within class distance to zero to improve the discrimination. Keep-It-Simple-and-Straightforward MEtric (KISSME) [75] learns the distance metric via a maximum log-likelihood ratio test. Pairwise Constrained Component Analysise (PCCA) [83] uses a hinge loss objective function, while rPCCA [78] extends it by introducing a regularization term. In SVMML [120], a locally adaptive distance metric is learned in a soft-margin SVM framework. For all the kernel-based methods, we evaluated 4 different kernels: linear (ℓ), chi-square (χ^2), chi-square-rbf (R_{χ^2}) and exponential (exp).

Table 5.4: Evaluated metric learning methods

Metric	Source	Metric	Source
FDA [118]	AE 1936	SVMML [120]	CVPR 13
MFA [119]	PAMI 07	kMFA [78]	ECCV 14
KISSME [75]	CVPR12	rPCCA [78]	ECCV 14
PCCA [83]	CVPR 12	kLFDA [78]	ECCV 14
kPCCA [83]	CVPR 12	XQDA [42]	CVPR 15
LFDA [77]	CVPR 13	NFST [79]	CVPR 16

5.2.3 Implementation Details

Prior to feature extraction, all bounding boxes are normalized to 128×64 pixels. In LDFV, the number of Gaussians for the GMM is set to 16. The number of bins in the color histogram for HistLBP and ELF is set to 16, and we use RGB as the color space in GOG. In metric learning, we set the subspace dimension to 40 and the negative-to-positive pair ratio to construct the training data to 10.

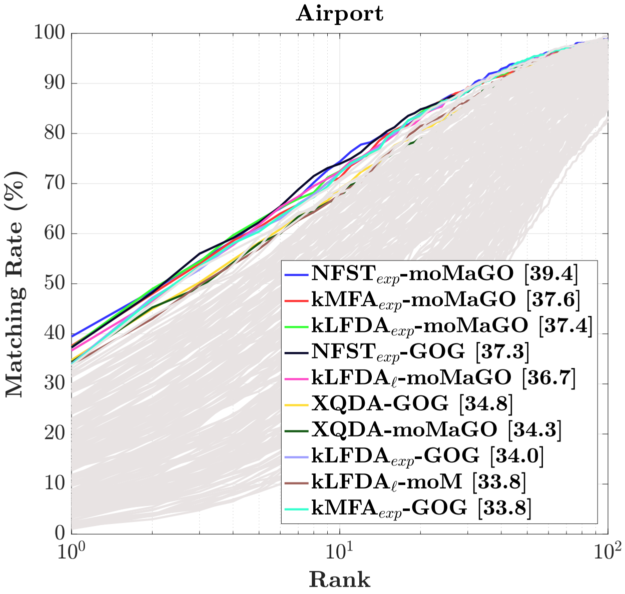


Figure 5.4: CMC curves for the benchmark on the Airport dataset. The top 10 performing algorithms are shown in color and the rest are shown in gray. Numbers in the brackets in the legend are the corresponding mAP value

5.2.4 Results and discussion

All experimental results are shown in Table 5.5 and 5.6 and the corresponding CMC curves are shown in Figure 5.4 and 5.5. For both benchmark datasets, the combination of moMaGO and NFST achieves the best performance. We will discuss the two factors feature selection and metric learning methods separately.

First, moMaGO feature dominates the performance in both datasets. Specifically, in the Airpot dataset, it obtains the second best or better results with all possible metric learning methods except KISSME. In the DukeMTMC4ReID dataset, it performs the best with all possible metric

Table 5.5: Rank 1 results from all feature/method combinations for Airport dataset. The best result for each metric learning is marked in red and the second best is in blue. The best combination across the whole dataset is mark in bold red.

Airport	Kernel	HistLBP	ELF	LDFV	gBiCov	SDC	LOMO	WHOS	GOG	moM	moMaGO
FDA		15.7	16.2	15.2	13.9	14.4	16.9	15.7	20.1	25.7	29.2
LFDA		16.0	17.8	14.7	13.6	15.2	17.8	17.2	20.8	26.8	29.8
kLFDA	ℓ	21.2	22.4	21.3	12.7	16.0	28.3	26.1	29.8	33.8	36.7
	χ^2	22.8	26.6	-	12.9	20.1	29.1	26.3	-	-	-
	R_{χ^2}	24.5	25.9	-	10.0	19.5	26.4	24.9	-	-	-
	exp	22.9	25.3	26.0	14.0	18.2	31.9	26.6	34.0	29.9	37.4
PCCA		5.3	2.8	4.6	5.9	2.0	4.1	2.3	7.0	10.9	11.5
kPCCA	ℓ	3.6	8.9	4.6	7.3	4.4	6.2	7.2	7.3	10.9	13.0
	χ^2	6.9	5.3	-	7.8	3.8	5.4	9.7	-	-	-
	R_{χ^2}	6.4	8.7	-	4.1	6.7	7.3	11.5	-	-	-
	exp	8.8	10.0	11.1	9.6	6.7	13.4	13.0	18.7	14.6	17.4
rPCCA	ℓ	3.3	7.8	4.5	7.3	6.0	5.2	7.7	8.2	7.8	8.9
	χ^2	5.9	8.4	-	7.7	3.5	5.5	10.4	-	-	-
	R_{χ^2}	5.5	10.3	-	4.0	3.2	7.3	11.0	-	-	-
	exp	9.8	10.0	11.1	8.5	6.3	13.7	13.5	18.0	13.8	20.4
MFA		13.9	10.0	3.6	11.5	6.5	8.3	4.6	15.8	14.6	21.7
kMFA	ℓ	17.2	21.3	19.0	11.8	13.8	21.6	24.0	29.9	31.4	32.5
	χ^2	18.1	25.2	-	10.8	15.8	22.1	24.4	-	-	-
	R_{χ^2}	15.3	24.1	-	8.8	14.5	23.1	27.0	-	-	-
	exp	17.0	21.5	25.9		14.2	26.6	26.5	33.8	33.1	37.6
NFST	ℓ	2.5	2.4	1.5	1.2	1.8	3.4	2.2	5.6	4.1	13.3
	χ^2	4.0	4.8	-	3.4	5.3	2.7	7.6	-	-	-
	R_{χ^2}	15.2	18.8	-	4.4	16.5	19.5	22.9	-	-	-
	exp	21.4	24.6	17.5	14.0	19.4	32.2	29.5	37.3	31.4	**39.4**
svmml		20.0	18.8	24.8	4.1	18.7	20.6	17.7	25.1	28.3	33.7
KISSME		5.0	6.9	7.6	8.5	6.3	5.4	9.9	4.9	2.3	4.4
XQDA		18.0	17.3	21.3	5.8	13.3	28.5	15.2	34.8	28.6	34.3

Table 5.6: Rank 1 results from all feature/method combinations for DukeMTMC4ReID dataset. The best result for each metric learning is marked in red and the second best is in blue. The best combination across the whole dataset is mark in bold red.

DukeMTMC4ReID	Kernel	HistLBP	ELF	LDFV	gBiCov	SDC	LOMO	WHOS	GOG	moM	moMaGO
FDA		19.1	22.2	23.7	17.2	22.8	20.7	23.7	26.3	32.7	35.6
LFDA		19.2	22.6	25.5	18.8	22.9	22.1	24.9	27.6	34.2	36.6
kLFDA	ℓ	17.0	18.8	31.9	17.0	21.3	31.1	30.8	42.3	41.7	49.5
	χ^2	27.3	27.3	-	14.3	27.8	31.4	34.2	-	-	-
	R_{χ^2}	32.2	29.4	-	14.4	29.7	31.2	34.9	-	-	-
	exp	28.9	26.4	37.3	15.0	25.1	34.4	33.9	45.2	45.5	52.2
PCCA		15.8	17.3	17.3	11.4	17.4	17.7	21.1	22.5	29.9	33.3
kPCCA	ℓ	13.1	14.7	22.7	10.5	15.0	25.1	22.5	33.2	29.1	34.2
	χ^2	20.4	21.4	-	10.3	21.0	24.2	25.2	-	-	-
	R_{χ^2}	23.4	23.4	-	12.3	23.7	26.5	28.4	-	-	-
	exp	18.4	17.7	25.0	13.4	19.5	27.6	28.0	34.1	30.7	35.9
rPCCA	ℓ	13.5	14.8	22.9	10.4	15.8	25.1	22.2	33.0	28.5	34.4
	χ^2	20.2	21.4	-	10.2	21.0	24.1	25.2	-	-	-
	R_{χ^2}	23.2	22.8	-	12.4	23.7	26.2	28.5	-	-	-
	exp	18.7	17.8	26.0	13.5	18.6	27.3	28.0	35.9	33.0	38.5
MFA		17.4	18.0	20.4	17.3	16.6	16.1	15.9	13.6	13.6	11.0
kMFA	ℓ	22.1	22.4	32.5	14.6	22.1	30.8	30.3	41.5	41.1	48.7
	χ^2	31.2	30.4	-	12.9	30.1	31.8	34.2	-	-	-
	R_{χ^2}	34.8	32.2	-	16.2	31.2	31.3	34.8	-	-	-
	exp	30.4	26.7	37.0	15.1	26.3	34.7	34.9	45.7	45.7	52.2
NFST	ℓ	0.1	0.1	0.1	0.1	0.1	0.4	0.0	0.0	0.4	1.9
	χ^2	1.3	2.8	-	1.0	1.8	1.8	2.4	-	-	-
	R_{χ^2}	5.3	5.5	-	1.6	8.2	19.6	12.7	-	-	-
	exp	33.0	17.2	40.0	14.5	25.5	43.4	42.6	49.9	51.2	**56.9**
svmml		4.1	9.6	33.7	2.2	15.3	11.4	19.5	26.9	22.4	34.3
KISSME		3.0	6.2	4.6	12.3	3.9	3.6	9.2	2.1	0.6	1.4
XQDA		10.2	21.3	28.1	1.9	21.6	28.8	29.2	35.7	34.1	48.7

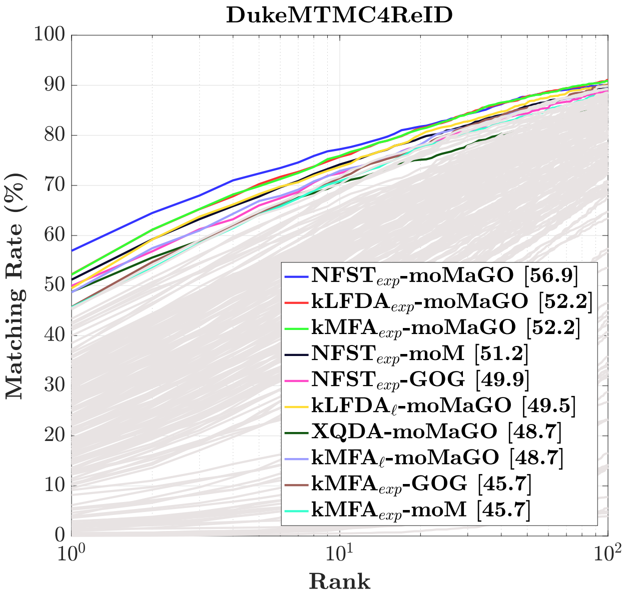


Figure 5.5: CMC curves for the benchmark on the DukeMTMC4ReID dataset. The top 10 performing algorithms are shown in color and the rest are shown in gray. Numbers in the brackets in the legend are the corresponding mAP value

learning methods except KISSME. Among the other features, either GOG or moM takes the second best performance, except for KISSME in Airport dataset and MFA, SVMML and KISSME in DukeMTMC4ReID dataset. This observation confirms the power of statistical moment modeling in feature representation for re-ID. Following them, the second tier features are LOMO and WHOS. LOMO also takes advantage of the hierarchical structure by aggregating the local patch representations along the same height. WHOS weights each pixel with a spatial Gaussian mask, which is similar to the patch weights in GOG and moM. This suggests the hierarchical structure with background subtraction will benefit the feature extraction in re-ID.

Next, we analyze the performance of different metric learning methods. NFST with exponential kernel achieves the best performance in both benchmark datasets. kLFDA and kMFA with exponential kernel obtain the second best results. It is interesting to note that all these three algorithms learn the distance metric by solving some form of generalized eigenvalue decomposition problems, similar to traditional Fisher discriminant analysis. While kLFDA and kMFA directly employ Fisher-type objective functions, NFST uses the Foley-Shannon transform [121], which is very closely related to the Fisher discriminant analysis. This suggests that the approach of formulating discriminant objective functions in terms of data scatter matrices is most suitable to the re-id problem.

Chapter 6

MoNet: Moment Embedding Network

Bilinear pooling has been recently proposed as a feature encoding layer, which can be used after the convolutional layers of a deep network, to improve performance in multiple vision tasks. Different from conventional global average pooling or fully connected layers, bilinear pooling gathers 2nd order information in a translation invariant fashion. However, a serious drawback of this family of pooling layers is their dimensionality explosion. Approximate pooling methods with compact properties have been explored towards resolving this weakness. Additionally, recent results have shown that significant performance gains can be achieved by adding 1st order information and applying matrix normalization to regularize unstable higher order information. However, combining compact pooling with matrix normalization and other order information has not been explored until now. In this section, we unify bilinear pooling and the global Gaussian embedding layers through the empirical moment matrix. In addition, we propose a novel sub-matrix square-root layer, which can be used to normalize the output of the convolution layer directly and mitigate the dimensionality problem with off-the-shelf compact pooling methods.

6.1 Introduction

Embedding local representations of an image to form a feature that is representative yet invariant to nuisance noise is a key step in many computer vision tasks. Before the phenomenal success of deep convolutional neural networks (CNN) [25], researchers tackled this problem with handcrafted consecutive independent steps. Remarkable works include HOG [122], SIFT [16], covariance descriptor [23], VLAD [21], Fisher vector [9] and bilinear pooling [27]. Although CNNs are trained from end to end, they can be also viewed as two parts, where the convolutional layers are

Table 6.1: Comparison of 2nd order statistic information based neural networks. BCNNonly has 2nd order information and does not use matrix normalization. Both improved BCNN (iBCNN) and G2DeNet take advantage of matrix normalization but suffer from large dimensionality because they use the square-root of a large pooled matrix. Our proposed MoNet, with the help of a novel sub-matrix square-root layer, can normalize the local features directly and reduce the final representation dimension significantly by substituting the fully bilinear pooling with compact pooling.

	1st order	Matrix	Compact
	moment	normalization	capacity
BCNN [28, 31]	✗	✗	✓
iBCNN [35]	✗	✓	✗
G^2DeNet [11]	✓	✓	✗
MoNet	✓	✓	✓

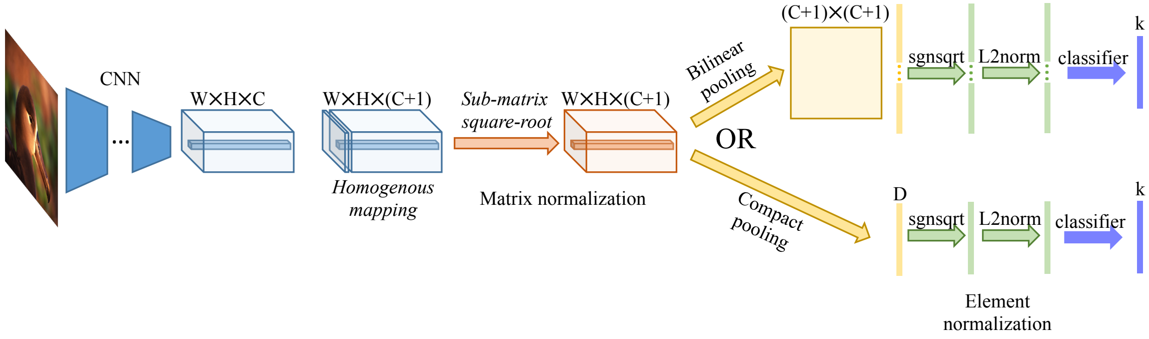


Figure 6.1: Architecture of the proposed moments-based network **MoNet**. With the proposed sub-matrix square-root layer, it is possible to perform matrix normalization before bilinear pooling or further apply compact pooling to reduce the dimensionality dramatically without undermining performance.

feature extraction steps and the later fully connected (FC) layers are an encoding step. Several works have been done to explore substituting the FC layers with conventional embedding methods in both two-stage fashion [123, 124] and end-to-end trainable way [28, 30].

Bilinear CNN (BCNN) was first proposed by Lin *et al.* [28] to pool the second order statistics information across the spatial locations. Bilinear pooling has been proven to be successful in many tasks, including fine-grained image classification [34, 31], large-scale image recognition [125], segmentation [30], visual question answering [126, 127], face recognition [128] and artistic style reconstruction [129]. Wang *et al.* [11] proposed to also include the 1st order information by using a Gaussian embedding in G^{2}DeNet. It has been shown that the normalization method is also critical to these CNNs' performance. Two normalization methods have been proposed for the bilinear pooled matrix, $\mathbf{M} = \frac{1}{n}\mathbf{X}^T\mathbf{X}$, where $\mathbf{X} \in \mathbb{R}^{n \times C}$ represents the local features. On one hand, because $\mathbf{M}$ is Symmetric Positive Definite (SPD), Ionescu *et al.* [30] proposed to apply matrix-logarithm to map the SPD matrices from the Riemannian manifold to an Euclidean space, followed by $\log(\mathbf{M}) = \mathbf{U}_M \log(\mathbf{S}_M)\mathbf{U}_M^T$ with $\mathbf{M} = \mathbf{U}_M\mathbf{S}_M\mathbf{U}_M^T$. On the other hand, [11, 35] proposed matrix-power to scale $\mathbf{M}$ non-linearly with $\mathbf{M}^p = \mathbf{U}_M\mathbf{S}_M^p\mathbf{U}_M^T$. In both works, matrix-power was shown to have better performance and numerically stability than the matrix-logarithm. In addition, Li *et al.* [125] provided theoretical support on the superior performance of matrix-power normalization in solving a general large-scale image recognition problem.

A critical weakness of the above feature encoding is the extremely high dimensionality of the encoded features. Due to the tensor product[1], the final feature dimension is C^2 where C is the number of feature channels of the last convolution layer. Even for relatively low $C = 512$ as in VGG-16 [2], the dimensionality of the final feature is already more than $262K$. This problem can be alleviated by using random projections [31], tensor sketching [31, 130], and the low rank property [34]. However, because the matrix-power normalization layer is applied on the pooled matrix $\mathbf{M}$, it is non-trivial to combine matrix normalization and compact pooling to achieve better performance and reduce the final feature dimensions at the same time.

In this paper, we propose a new architecture, MoNet, that integrates matrix-power normalization with Gaussian embedding. To this effect, we re-write the formulation of G^{2}DeNet using the tensor product of the homogeneous padded local features to align it with the architecture of BCNN so that the Gaussian embedding operation and bilinear pooling are decoupled. Instead of working on the bilinear pooled matrix $\mathbf{M}$, we derive the sub-matrix square-root layer to perform the matrix-power

[1] We show that the Gaussian embedding can be written as a tensor product in sec. 6.3.2.1 In the following sections, we will use tensor product and bilinear pooling interchangeably.

normalization directly on the (in-)homogeneous local features. With the help of this novel layer, we can take advantage of compact pooling to approximate the tensor product, but with much fewer dimensions.

The main contributions of this work are three-fold:

- We unify the G^{2}DeNet and bilinear pooling CNN using the empirical moment matrix and decouple the Gaussian embedding from bilinear pooling.
- We propose a new sub-matrix square-root layer to directly normalize the features before the bilinear pooling layer, which makes it possible to reduce the dimensionality of the representation using compact pooling.
- We derive the gradient of the proposed layer using matrix back propagation, so that the whole proposed *moments embedding network* **“MoNet”** architecture can be optimized jointly.

6.2 Related Work

Bilinear pooling was proposed by Tenenbaum *et al.* [24] to model two-factor structure in images to separate style from content. Lin *et al.* [28] introduced it into a convolutional neural network as a pooling layer and improved it further by adding matrix power normalization in their recent work [35]. Wang *et al.* [11] proposed G^{2}DeNet with Gaussian embedding, followed by matrix normalization to incorporate 1st order moment information and achieved the state-of-the-art performance. In a parallel research track, low dimension compact approximations of bilinear pooling have been also explored. Gao *et al.* [31] bridged bilinear pooling with a linear classifier with a second order polynomial kernel by adopting the off-the-shelf kernel approximation methods Random MacLaurin [32] and Tensor Sketch [33] to pool the local features in a compact way. Cui [130] generalized this approach to higher order polynomials with Tensor Sketch. By combining with bilinear SVM, Kong *et al.* [34] proposed to impose a low-rank constraint to reduce the number of parameters. However, none of these approaches can be easily integrated with matrix normalization because of the absence of a bilinear pooled matrix.

Lasserre *et al.* [131] proposed to use the empirical moment matrix formed by explicit in-homogeneous polynomial kernel basis for outlier detection. Sznaier *et al.* [132] improved the performance for the case of data subspaces, by working on the singular values directly. In [117], the empirical moments matrix was applied as a feature embedding method for the person re-identification problem and it was shown that the Gaussian embedding [44] is a special case when the moment matrix order equals to 1. However, both of these works focus on a conventional pipeline and did not bring moments to modern CNN architectures.

Ionescu *et al.* [30] introduced the theory and practice of matrix back-propagation for training CNNs, which enable structured matrix operations in deep neural networks training. Both [35] and [11] used it to derive the back-propagation of the matrix square-root and matrix logarithm for a symmetric matrix. Li *et al.* [125] applied a generalized p-th order matrix power normalization instead of the square-root. However, in our case, since we want to apply the matrix normalization directly on a non-square local feature matrix, we cannot plug-in the equation directly from previous works.

6.3 MoNet Architecture

The overview of the proposed MoNet architecture is shown in Fig. 6.1. For an input image $\mathbf{I}$, the output of the last convolution layer after the ReLU, $\mathbf{X}$, consists of local features $\mathbf{x}_i$, across spatial locations $i = 1, 2, \ldots, n$. Then, we introduce a homogeneous mapping (HM) layer to disentangle the tensor product operator. After that, a novel sub-matrix square-root (Ssqrt) layer is applied to directly normalize the feature vector before the tensor product. Finally, a compact bilinear pooling layer pools all n features across all spatial locations, followed by an element-wise square-root regularization and ℓ_2 normalization before the final fully-connected layer. Next, we will detail the design of each block.

6.3.1 Homogeneous mapping layer

Since the global Gaussian embedding layer used in G^{2}DeNet entangles the tensor product operator, one cannot directly incorporate compact bilinear pooling. With the help of the proposed HM layer, we can re-write the Gaussian embedding layer with a HM layer followed by a tensor product, as explained next.

Assume $\mathbf{X} \in \mathbb{R}^{n \times C}$, corresponding to n features with dimension C and $n > C$, mean μ and covariance $\mathbf{\Sigma}$. The homogeneous mapping of $\mathbf{X}$ is obtained by padding $\mathbf{X}$ with an extra dimension set to 1. For the simplicity of the following layers, instead of applying the conventional bilinear pooling layer as in [28], we also divide the homogeneous feature by the square-root of the number of samples. Then, the forward equation of the homogeneous mapping layer is:

$$\tilde{\mathbf{X}} = \frac{1}{\sqrt{n}}[\mathbf{1}|\mathbf{X}] \in \mathbb{R}^{n \times (C+1)} \tag{6.1}$$

The tensor product of $\tilde{\mathbf{X}}$ can be written as

$$\mathbf{M} = \tilde{\mathbf{X}}^T\tilde{\mathbf{X}} = \begin{bmatrix} \mathbf{1} & \mu \\ \mu^T & \frac{1}{n}\mathbf{X}^T\mathbf{X} \end{bmatrix} \tag{6.2}$$

where $\mu = \frac{1}{n}\sum_1^n \mathbf{X}$. Since $\frac{1}{n}\mathbf{X}^T\mathbf{X} = \mathbf{\Sigma} + \mu^T\mu$, Eq. 6.2 is the Gaussian embedding method used in G^{2}DeNet [11]. One can also show that the conventional bilinear pooling layer is equal to the tensor product of the in-homogeneous feature matrix.

6.3.2 Sub-matrix square-root layer

Matrix normalization in iBCNN and G^{2}DeNet requires the computation of the singular value decomposition (SVD) of the output of the tensor product, which prevents the direct use of compact bilinear pooling. We will address this issue by incorporating a novel layer, named sub-matrix square-root (Ssqrt) layer, to perform the equivalent matrix normalization before the tensor product. This choice is supported by experimental results in [11, 35] showing that the matrix square-root normalization is better than the matrix logarithm normalization for performance and training stability.

6.3.2.1 Forward propagation

Recall that given the SVD of a SPD matrix, $\mathbf{Q} = \mathbf{U}_Q\mathbf{S}_Q\mathbf{U}_Q^T$, the square root of $\mathbf{Q}$ is defined as

$$\mathbf{Q}^{\frac{1}{2}} = \mathbf{U}_Q\mathbf{S}_Q^{\frac{1}{2}}\mathbf{U}_Q^T \tag{6.3}$$

where $\mathbf{S}_Q^{\frac{1}{2}}$ is computed by taking the square root of its diagonal elements.

Consider now the SVD of $\tilde{\mathbf{X}} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. Then, we have

$$\mathbf{M} = \tilde{\mathbf{X}}^T\tilde{\mathbf{X}} = \mathbf{V}\mathbf{S}^T\mathbf{U}^T\mathbf{U}\mathbf{S}\mathbf{V}^T \tag{6.4}$$

and since $\mathbf{U}^T\mathbf{U} = \mathbf{I}$ and $\mathbf{S}^T\mathbf{S}$ is a square matrix:

$$\mathbf{M}^{\frac{1}{2}} = \mathbf{V}(\mathbf{S}^T\mathbf{S})^{\frac{1}{2}}\mathbf{V}^T \tag{6.5}$$

Note that $\mathbf{S} \in \mathbb{R}^{n\times(C+1)}, n > C+1$ and hence its square root is not well defined. We introduce a helper matrix $\mathbf{A}$ to keep all non-zero singular values in $\mathbf{S}$ as follows:

$$\mathbf{S} = \mathbf{A}\tilde{\mathbf{S}}, \mathbf{A} = [\mathbf{I}_{C+1}|\mathbf{0}]^T \tag{6.6}$$

where $\tilde{\mathbf{S}} \in \mathbb{R}^{(C+1)\times(C+1)}$ is a square diagonal matrix and $\mathbf{I}_{C+1}$ is the $(C+1)\times(C+1)$ identity matrix. Substituting Eq. (6.6) in Eq. (6.5), we have

$$\mathbf{M}^{\frac{1}{2}} = \mathbf{V}(\tilde{\mathbf{S}}\mathbf{A}^T\mathbf{A}\tilde{\mathbf{S}})^{\frac{1}{2}}\mathbf{V}^T = \mathbf{V}\tilde{\mathbf{S}}^{\frac{1}{2}}\tilde{\mathbf{S}}^{\frac{1}{2}}\mathbf{V}^T \tag{6.7}$$

since $\mathbf{A}^T\mathbf{A} = \mathbf{I}_{C+1}$. To keep the same number of samples for the input and output of this layer, we finally re-write Eq. (6.5) in the following tensor product format:

$$\mathbf{M}^{\frac{1}{2}} = \mathbf{Y}^T\mathbf{Y} \tag{6.8}$$

where the output $\mathbf{Y}$ is defined as $\mathbf{Y} = \mathbf{A}\tilde{\mathbf{S}}^{\frac{1}{2}}\mathbf{V}^T$, allowing us to perform matrix normalization directly on the features $\tilde{\mathbf{X}}$.

Note that because in most modern CNNs, n cannot be much greater than C and the features after ReLU tend to be sparse, $\tilde{\mathbf{X}}$ is usually rank deficient. Therefore, we only use the non-zero singular values and singular vectors. Then, the forward equation of the sub-matrix square-root layer can be written as

$$\mathbf{Y} = \mathbf{A}_{:,1:e}\tilde{\mathbf{S}}^{\frac{1}{2}}_{1:e}\mathbf{V}^T_{:,1:e} \tag{6.9}$$

where e is the index of the smallest singular value greater than ϵ [2].

6.3.2.2 Backward propagation

We will follow the matrix back propagation techniques proposed by Ionescu *et al.* [30] to derive the equation of the back propagation path for the sub-matrix square-root layer.

For a scalar loss $L = f(\mathbf{Y})$, we assume $\frac{\partial L}{\partial \mathbf{Y}}$ is available when we derive the back propagation. Let $\tilde{\mathbf{X}} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ and $\mathbf{U} \in \mathbb{R}^{n\times n}$. We can form $\mathbf{U}$ using block decomposition as $\mathbf{U} = [\mathbf{U}_1|\mathbf{U}_2]$ with $\mathbf{U}_1 \in \mathbb{R}^{n\times(C+1)}$ and $\mathbf{U}_2 \in \mathbb{R}^{n\times(n-C-1)}$. The partial derivatives between a given scalar loss L and $\tilde{\mathbf{X}}$ are

$$\begin{aligned}\frac{\partial L}{\partial \tilde{\mathbf{X}}} =& \mathbf{D}\mathbf{V}^T + \mathbf{U}(\frac{\partial L}{\partial \mathbf{S}} - \mathbf{U}^T\mathbf{D})_{diag}\mathbf{V}^T + \\ & 2\mathbf{U}\mathbf{S}(\mathbf{K}^T \circ \left(\mathbf{V}^T(\frac{\partial L}{\partial \mathbf{V}} - \mathbf{V}\mathbf{D}^T\mathbf{U}\mathbf{S})\right)_{sym}\mathbf{V}^T\end{aligned} \tag{6.10}$$

where $\circ$ represents element-wise product, $(\mathbf{Q})_{sym} \doteq \frac{1}{2}(\mathbf{Q}^T + \mathbf{Q})$ and

$$\mathbf{D} = \left(\frac{\partial L}{\partial \mathbf{U}}\right)_1 \tilde{\mathbf{S}}^{-1} - \mathbf{U}_2 \left(\frac{\partial L}{\partial \mathbf{U}}\right)_2^T \mathbf{U}_1\tilde{\mathbf{S}}^{-1} \tag{6.11}$$

[2] We will omit the subscript in the following for a concise notation

$$\mathbf{K}_{ij} = \begin{cases} \frac{1}{s_i^2 - s_j^2} & i \neq j \\ 0 & i = j \end{cases} \tag{6.12}$$

From Eq. 6.9, we can compute the variation of $\mathbf{Y}$ as

$$d\mathbf{Y} = \frac{1}{2}\mathbf{A}\tilde{\mathbf{S}}^{-\frac{1}{2}} d\tilde{\mathbf{S}}\mathbf{V}^T + \mathbf{A}\tilde{\mathbf{S}}^{\frac{1}{2}} d\mathbf{V}^T \tag{6.13}$$

Based on the chain rule, the total variation can be written as

$$\frac{\partial L}{\partial \mathbf{Y}} : d\mathbf{Y} = \frac{1}{2}\frac{\partial L}{\partial \mathbf{Y}} : \mathbf{A}\tilde{\mathbf{S}}^{-\frac{1}{2}} d\tilde{\mathbf{S}}\mathbf{V}^T + \frac{\partial L}{\partial \mathbf{Y}} : \mathbf{A}\tilde{\mathbf{S}}^{\frac{1}{2}} d\mathbf{V}^T \tag{6.14}$$

where : denotes the inner-product. After re-arrangement with the rotation properties of inner-product, we re-write the above equation as

$$\frac{\partial L}{\partial \mathbf{Y}} : d\mathbf{Y} = \frac{1}{2}\tilde{\mathbf{S}}^{-\frac{1}{2}}\mathbf{A}^T\frac{\partial L}{\partial \mathbf{Y}}\mathbf{V} : d\tilde{\mathbf{S}} + \tilde{\mathbf{S}}^{\frac{1}{2}}\mathbf{A}^T\frac{\partial L}{\partial \mathbf{Y}} : d\mathbf{V}^T \tag{6.15}$$

Therefore, we have

$$\frac{\partial L}{\partial \mathbf{S}} = \mathbf{A}\frac{\partial L}{\partial \tilde{\mathbf{S}}} = \frac{1}{2}\mathbf{A}\tilde{\mathbf{S}}^{-\frac{1}{2}}\mathbf{A}^T\frac{\partial L}{\partial \mathbf{Y}}\mathbf{V} \tag{6.16}$$

$$\frac{\partial L}{\partial \mathbf{V}} = (\frac{\partial L}{\partial \mathbf{V}^T})^T = (\frac{\partial L}{\partial \mathbf{Y}})^T\mathbf{A}\tilde{\mathbf{S}}^{\frac{1}{2}} \tag{6.17}$$

Finally, substituting Eq. 6.16 and Eq. 6.17 into Eq. 6.10 and considering $\frac{\partial L}{\partial \mathbf{U}} = 0$, we have

$$\begin{aligned}\frac{\partial L}{\partial \tilde{\mathbf{X}}} =& \mathbf{U}\left(\frac{1}{2}\mathbf{A}\tilde{\mathbf{S}}^{-\frac{1}{2}}\mathbf{A}^T\frac{\partial L}{\partial \mathbf{Y}}\mathbf{V} + \right. \\ & \left. 2\mathbf{S}\left[\mathbf{K}^T \circ \left(\mathbf{V}^T\left(\frac{\partial L}{\partial \mathbf{Y}}\right)^T\mathbf{A}\tilde{\mathbf{S}}^{\frac{1}{2}}\right)\right]_{sym}\right)\mathbf{V}^T\end{aligned} \tag{6.18}$$

6.4 Compact pooling

Following the work in [31, 126], we adopt the Tensor Sketch (TS) method to approximate bilinear pooling due to it better performance and lower computational and memory cost. Building up on count sketch and FFT, one can generate a tensor sketch function s.t. $\langle TS_1(x), TS_2(y)\rangle \approx \langle x, y\rangle^2$, using Algorithm 2. The back-propagation of a TS layer is given by [31].

As shown in Table 6.2, with the techniques mentioned above, the proposed MoNet is capable to solve the problem with much less computation and memory complexity than the other BCNN based algorithms.

Algorithm 2 Tensor Sketch approximation pooling

```
Require: $x$, projected dimension $D$
Generate randomly selected (but fixed) two pairs of hash functions $h_t \in \mathbb{R}^D$ and $s_t \in \mathbb{R}^D$ where $t = 1, 2$ and $h_t(i)$, $s_t(i)$ are uniformly drawn from $\{1, 2, \cdots, D\}$ and $\{-1, +1\}$, respectively.
Define count sketch function $\Psi(x, h_t, s_t) = [\psi_1(x), \psi_2(x), \cdots, \psi_D(x)]^T$ where $\psi_j(x) = \sum_{i:h_t(i)=j} s_t(i)x_i$
Define $TS(x) = FFT^{-1}(FFT(\Psi(x, h_1, s_1) \circ (\Psi(x, h_2, s_2))))$ where $\circ$ denotes element-wise multiplication.
```

Table 6.2: Dimension, computation and memory information for different network architectures we compared in this paper. H, W and C represent the height, width and number of feature channels for the output of the final convolution layer, respectively. k and D denote the number of classes and projected dimensions for Tensor Sketch, respectively. Numbers inside brackets indicate the typical value when the corresponding network was evaluated with VGG-16 model [2] on a classification task with 1,000 classes. In this case, $H = W = 13, C = 512, k = 1000, D = 10000$ and all data was stored with single precision.

	BCNN [28]	iBCNN [35]	iBCNN TS	G^{2}DeNet [11]	MoNet	MoNet TS
Dimension	C^2 [262K]	C^2 [262K]	D [10K]	(C+1)^2 [263K]	(C+1)^2 [263k]	D [10k]
Parameter Memory	0	0	2C	0	0	2C
Computation	$O(HWC^2)$	$O(HWC^2)$	$O(HW(C + D\log D))$	$O(HWC^2)$	$O(HWC^2)$	$O(HW(C + D\log D))$
Classifier Memory	kC^2 [1000MB]	kC^2 [1000MB]	kD [40MB]	$k(C+1)^2$ [1004MB]	$k(C+1)^2$ [1004MB]	kD [40MB]

Chapter 7

Fine-grained Classification

The fine-grained classification problem aims to distinguish the sub-category classes. For instance, instead of classifying people, car and bird, it focus on discriminating the different models of cars and different species of birds. The differences on view-point, pose and illumination lead to large intra-class variances whereas the inter-class variance is usually very subtle. To alleviate these nuisance noise, a two-step scheme has been explored. It utilizes parts localization and alignment as the first step to reduce the intra-class variance and then trains a classifier in the second step. Part-based R-CNN [133] extends the R-CNN [134] to detect the parts. Spatial transformer networks [135] learns a model to transfer images to a canonical view prior classification. Recently, visual attention was adopted for the fine-grained classification problem to locate the discriminant region directly. Liu *et al.* [136] learns attention through the attribute description. Besides localizing the attentions, diversified visual attention network [137] can maximize the discrimination between the attentions at the same time. Except the above works, neural network with bilinear pooling layers dominate the performance in several widely used datasets.

In this section, we will perform experiments on three widely used fine-grained classification datasets to illustrate that our proposed architecture, MoNet, can achieve similar or better performance than with the state-of-art G^{2}DeNet. Furthermore, when combined with the compact pooling technique, MoNet obtains comparable performance with encoded features with 96% less dimensions. Aligned with other bilinear CNN based papers, we also evaluate the proposed MoNet with three widely used fine-grained classification datasets. The experimental setups and the algorithm implementation are described in detail in Sec. 7.1. Then, in Sec. 7.1.1, the experimental results on fine-grained classification are presented and analyzed.

Table 7.1: Basic statistics of the datasets used for evaluation

Datasets	# training	# testing	# classes
CUB [138]	5,994	5,794	200
Aircraft [139]	6,667	3,333	100
Cars [140]	8,144	8,041	196

7.1 Experimental setup

We evaluated MoNet on three widely used fine-grained classification datasets. Different from general object recognition tasks, fine-grained classification usually tries to distinguish objects at the sub-category level, such as different makes of cars or different species of birds. The main challenge of this task is the relatively large inter-class and relatively small intra-class variations.

In all experiments, the 13 convolutional layers of VGG-16 [2] are used as the local feature extractor, and their outputs are used as local appearance representations. These 13 convolution layers are trained with ImageNet [141] and fine tuned in our experiments with three fine-grained classification datasets.

7.1.1 Datasets

Caltech-UCSD birds (CUB) [138] contains 200 species, mostly north-American, of birds. Being consistent with other works, we also use the 2011 extension with doubled number samples.

FGVC-Aircraft Benchmark (Aircraft) [139] is a benchmark fine-grained classification dataset with different aircrafts with various models and manufacturers.

Stanford cars (Cars) [140] contains images of different classes of cars at the level of make, model and year.

We use the provided train/test splits for all three datasets. Detailed information is given in Table. 7.1 and Fig. 7.1 shows sample images.

7.1.2 Different pooling methods

Bilinear pooling (BCNN): The VGG-16 based BCNN [28] is utilized as the baseline pooling method, which applies the tensor product on the output of the conv$_{5\_3}$ layer with ReLU activation. The dimension of the final representation is $512 \times 512 \approx 262K$ and the number of the

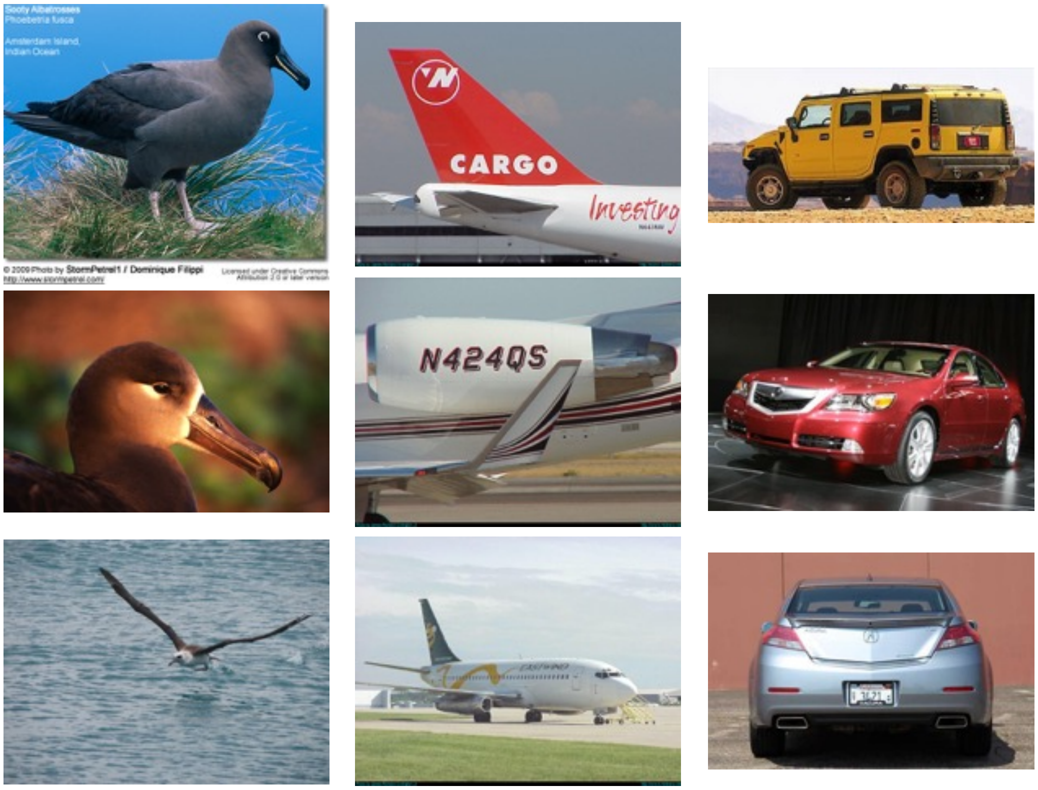

Sooty Albatrosses
Phoebetria fusca
Amsterdam Island,
Indian Ocean
© 2009 Photo by StormPetrel1 / Dominique Filippi
http://www.stormpetrel.com/
Licensed under Creative Commons
Attribution 2.0 or later version
CARGO
Investing
N424QS

Figure 7.1: Sample images from the fine-grained classification datasets. From left to right, each column corresponds to CUB, Aircraft and Cars, respectively.

Table 7.2: Experimental results for MoNet variants. Modifiers '2' and '**U**' indicate that only 2nd order moments are incorporated during feature embedding, and that no normalization was used, respectively. The abbreviations for proposed layers are denoted as: **SSqrt**: sub-matrix square root; **HM**: Homogeneous mapping. The best result in each column is marked in red.

New name	Missing layers	CUB		Airplane		Cars	
		Bilinear	TS	Bilinear	TS	Bilinear	TS
MoNet-2U	HM, Ssqrt	85.0	85.0	86.1	86.1	89.6	89.5
MoNet-2	HM	86.0	**85.7**	86.7	86.7	90.5	90.3
MoNet-U	Ssqrt	82.8	84.8	84.4	87.2	88.8	90.0
MoNet	-	**86.4**	**85.7**	**89.3**	**88.1**	**91.8**	**90.8**

linear classifier parameters is $k \times 262K$, where k is the number of classes. To be fair, the latest results from the authors' project page [142] are compared.

Improved bilinear pooling (iBCNN): Lin *et al.* [35] improved the original BCNN by adding the matrix power normalization after the bilinear pooling layer. We compare the results reported in [35] with VGG-16 as the back-bone network.

Global Gaussian distribution embedding (G^{2}DeNet): Instead of fully bilinear pooling, G^{2}DeNet pools the local features with a global Gaussian distribution embedding method, followed by a matrix square-root normalization. Since it includes the first order moment information, the dimension of the final feature is slightly greater than BCNN and iBCNN. The experiment results with "w/o BBox" configuration in [11] are compared in this paper.

Proposed moment embedding network (MoNet) and its variants: We implemented the proposed MoNet architecture with structure as shown in Fig. 6.1 and fine-tuned the whole network in an end-to-end fashion. When using bilinear pooling, the feature dimensionality, computation and memory complexity are the same as G^{2}DeNet. To evaluate the effectiveness of the proposed layers HM and Ssqrt, we also tested MoNet variants. Depending on the left-out layer, we can have four different variants in total. Modifiers '2' and '**U**' indicate that only 2nd order moments are incorporated during feature embedding, and that no normalization was used, respectively.

Tensor Sketch compact pooling (TS): When building the network with compact pooling, the TS layer [31] was added after the sub-matrix square-root layer. The projection dimension D was selected empirically for MoNet and its variants.

7.1.3 Implementation details

Using a large enough number of samples is important to estimate stable and meaningful statistical moment information. The input images are resized to 448×448 in all the experiments, which produces a $28 \times 28 \times 512$ local feature matrix after conv$_{5\_3}$ for each image. Following common practice [11, 130], we first resize the image with a fixed aspect-ratio, such as the shorter edge reaches to 448 and then utilized a center crop to resize the image to 448×448. During training, random horizontal flipping was applied as data augmentation. Different from [35] with VGG-M, no augmentation is applied during testing.

To avoid rank deficiency, the singular value threshold σ was set to 10^{-5} for both forward and backward propagation, which results in 10^{-10} for the singular value threshold of the tensor product matrix. The projected dimension in Tensor Sketch was fixed to $D = 10^4$, which satisfies $C < D \ll C^2$. For a smooth and stable training, we applied gradient clipping[143] to chop all gradients in the range $[-1, 1]$.

As suggested by [35, 11], all pooling methods were followed by an element-wise sign kept squre-root $\mathbf{y}_s = sign(\mathbf{y})\sqrt{\mathbf{y}}$ and ℓ_2 normalization $\mathbf{y}_n = \mathbf{y}_s/||\mathbf{y}_s||$. For the sake of a smooth training, the element-wise square-root is also applied on local appearence features [11].

The weights of the VGG-16 convolutional layers are pretrained on ImageNet classification dataset. We first warm-started by fine-tuning the last linear classifier for 300 epochs. Then, we fine-tuned the whole network end-to-end with the learning rate as 0.001 and batch size as 16. The momentum was set to 0.9 and the weight decay was set to 0.0005. Most experiments converged to a local optimum after 50 epochs.

The proposed MoNet was implemented with MatConvNet [144] and Matlab 2017a[1]. Because of the numerical instability of SVDs, as suggested by Ionescu *et al.* [30], the sub-matrix square-root layer was implemented on CPU with double precision. The whole network was fine-tuned on a Ubuntu PC with 64GB RAM and Nvidia GTX 1080 Ti.

7.2 Experimental results

In Table 7.2 and Table 7.3, the classification accuracy for each network is presented in a row. Bilinear and TS denote fully bilinear pooling and tensor sketch compact pooling, respectively.

[1] Code is available at `https://github.com/NEU-Gou/MoNet`

Table 7.3: Experimental results on fine-grained classification. Bilinear and TS represent fully bilinear pooling and Tensor Sketch compact pooling respectively. The best performance in each column is marked in red.

		CUB		Airplane		Car	
		Bilinear	TS	Bilinear	TS	Bilinear	TS
BCNN [28, 31]		84.0	84.0	86.9	87.2	90.6	90.2
MoNet-2U		85.0	85.0	86.1	86.1	89.6	89.5
iBCNN [35]		85.8	-	88.5	-	92.1	-
MoNet-2		86.0	**85.7**	86.7	86.7	90.5	90.3
G^{2}DeNet [11]		**87.1**	-	89.0	-	**92.5**	-
MoNet		86.4	**85.7**	**89.3**	**88.1**	91.8	**90.8**
Other higher order methods	KP [130]	-	**86.2**	-	86.9	-	**92.6**
	HOHC [145]	85.3		88.3		91.7	
State-of-the-art	MA-CNN [146]	86.5		89.9		92.8	

Comparison with different variants: The variants MoNet-2U, MoNet-2, and MoNet, when using bilinear pooling, are mathematically equivalent to BCNN, iBCNN, and G^{2}DeNet, respectively. Aligned with the observation in [35, 11], we also see a consistent performance gain for both MoNet and MoNet-2 by adding the normalization sub-matrix square root (SSqrt) layer. Specifically, MoNet-2 outperforms MoNet-2U by 0.6% to 1% with bilinear pooling and 0.6% to 0.8% with TS. Whereas MoNet outperforms MoNet-U by 3% to 4.9% with bilinear pooling and 0.8% to 0.9% with TS. This layer is more effective on MoNet than on MoNet-2. The reason for this, is that mixing different order moments may make the embedded feature numerically unstable but a good normalization helps overcome this issue. By adding the HM layer to incorporate 1st order moment information, MoNet can achieve better results consistently when compared to MoNet-2, in all datasets with both bilinear and compact pooling. Note that MoNet-U performs worse than MoNet-2U, which actually illustrates the merit of a proper normalization.

Comparison with different architectures: Consistent with [35], matrix normalization improves the performance by 1-2% on all three datasets. Our equivalent MoNet-2 achieves slightly better classification accuracy (0.2%) on CUB dataset but performs worse on Airplane and Car datasets when compared with iBCNN. We believe that this is due to the different approaches used to deal with rank deficiency. In our implementation, the singular value is hard thresholded as shown in Eq. 6.9, while iBCNN [35] dealt with the rank deficiency by adding 1 to all the singular values, which is a relatively very small number compared to the maximum singular value (10^6). By adding the 1st order moment information, G^{2}DeNet outperforms iBCNN by around 1%, on all three datasets. By re-writing the Gaussian embedding with tensor product of the homogeneous padded local features, our proposed MoNet can obtain similar or slightly better classification accuracy when comparing against G^{2}DeNet. Specifically, the classification accuracy of MoNet is 0.3% higher on Airplane dataset, but 0.7% lower on both CUB and Car datasets.

Comparison with fully bilinear pooling and compact pooling: As shown in [31], compact pooling can achieve similar performance compared to BCNN, but with only 4% of the dimensionality. We also see a similar trend in MoNet-2U and MoNet-2. The classification accuracy difference between the bilinear pooling and compact pooling version is less than 0.3% on all three datasets. However, the performance gaps are relatively greater when we compare the different pooling schemes on MoNet. Bilinear pooling improve the classification accuracy by 0.7%, 1.2% and 1% than compact pooling on CUB, Airplane and Car datasets, respectively. However, with compact pooling, the dimensionality of the final representation is 96% smaller. Although the final fully bilinear pooled representation dimensions of MoNet-2 and MoNet are roughly the same, MoNet utilizes more different order moments, which requires more count sketch projections to approximate it. Thus, when fixing $D = 10,000$ for both MoNet-2 and MoNet, the performance of MoNet with compact pooling degraded. However, MoNet TS still out-performs MoNet-2 TS by 1.4% and 0.5% on the Airplane and Car datasets, respectively.

Comparison with other methods: [130] and [145] are two other recent works that also take into account higher order statistic information. Cui *et al.* [130] applied Tensor Sketch repetitively to approximate up to 4th order explicit polynomial kernel space in a compact way. They obtained better results for CUB and Car datasets compared against other compact pooling results, but notably worse (1.2%) on the Airplane dataset. This may be due to two factors. First, directly utilizing higher order moments without proper normalization leads to numerically instabilities. Second, approximating higher order moments with limited number of samples is essentially an ill-posed problem. Cai *et al.* [145] only utilize higher order self-product terms but not the interaction terms, which leads to worse performance in all three datasets. Finally, the state-of-the-art MA-CNN [146] achieves slightly better results on Airplane and Car datasets.

Chapter 8

Conclusion

In this dissertation, we proposed a feature embedding method with empirical moment matrix. By incorporating higher order moments information, the proposed method can approximate arbitrary distributions efficiently. We integrated this embedding method in conventional feature extraction pipelines and tested it on re-ID. The novel feature moM generalizes the Gaussian assumption used in previous work by using the empirical moment matrix. The extensive experimental results on five datasets illustrate the effectiveness of this feature and of combining it with GOG feature sets, achieving a new state-of-the-art performance for three datasets. Directly comparing re-ID algorithms reported in the literature has become difficult since a wide variety of features, experimental protocols, and evaluation metrics are employed. To address this need, we presented two large-scale datasets collected with real surveillance camera networks and established a reliable benchmark with single-shot re-ID algorithms. The experimental results illustrate the effectiveness of the statistical empirical moment matrix feature in re-ID and the discussions on the benchmark results shed light on the future design for feature extraction and metric learning algorithms in re-ID.

We also fused the empirical moment matrix embedding into the modern CNN architecture with novel layers. We reformulated the Gaussian embedding using the empirical moment matrix and decoupled the bilinear pooling step out. With the help of a novel sub-matrix square-root layer, our proposed network MoNet can take advantages of different order moments, matrix normalization as well as compact pooling. Experiments on three widely used fine-grained classification datasets demonstrate that MoNet can achieve similar or better performance when comparing with G2DeNet and retain comparable results with only 4% of the feature dimensions.

There are several open problems and possible extensions following this dissertation. Future works include:

- *Compact moment representation:* Since the dimensionality of the empirical moment matrix grows exponentially with the order and local descriptor size, designing a compact representation for it is an emerging direction.

- *Moment of moment:* For moM feature, instead of using an on-manifold mean, with the help of compact moment representation, one can build the empirical moment matrix on the second level as well to model the global distribution of the descriptors of the patches.

- *Deep benchmark:* Since the deep neural network models dominate the re-ID literature in recent years, extending the re-ID benchmark to include state-of-the-art deep neural network models will make the benchmark more comprehensive.

- *Higher order MoNet:* Given the promising results of applying higher order moment matrix in conventional feature extraction pipelines, generalizing the homogeneous mapping layer to incorporate higher order information can be one possible direction.

Bibliography

[1] "DukeMTMC project," http://vision.cs.duke.edu/DukeMTMC/details.html, accessed: 2017-03-22.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[3] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *ECCV*, 2006.

[4] H. Q. Minh and V. Murino, "Covariances in computer vision and machine learning," *Synthesis Lectures on Computer Vision*, vol. 7, no. 4, pp. 1–170, 2017.

[5] L. Gong, T. Wang, and F. Liu, "Shape of gaussians as feature descriptors," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.* IEEE, 2009, pp. 2366–2371.

[6] Z. Huang, R. Wang, S. Shan, X. Li, and X. Chen, "Log-euclidean metric learning on symmetric positive definite manifold with application to image set classification," in *International conference on machine learning*, 2015, pp. 720–729.

[7] H. Nakayama, T. Harada, and Y. Kuniyoshi, "Global gaussian approach for scene categorization using information geometry," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 2010, pp. 2336–2343.

[8] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 2. IEEE, 2004, pp. 28–31.

[9] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," *Computer Vision–ECCV 2010*, pp. 143–156, 2010.

[10] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato, "Hierarchical gaussian descriptor for person re-identification," in *CVPR*, 2016.

[11] Q. Wang, P. Li, and L. Zhang, "G2denet: Global gaussian distribution embedding network and its application to visual recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[12] "The british machine vision association and society for pattern recognition," http://www.bmva.org/visionoverview, accessed: 2018-04-13.

[13] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[14] L. Zheng, Y. Yang, and Q. Tian, "SIFT meets CNN: A decade survey of instance retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. [Online]. Available: https://doi.org/10.1109/tpami.2017.2709749

[15] C. Harris and M. Stephens, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15, no. 50. Citeseer, 1988, pp. 10–5244.

[16] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

[17] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[18] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Computer Vision (ICCV), 2011 IEEE international conference on*. IEEE, 2011, pp. 2564–2571.

[19] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *T-PAMI*, vol. 32, no. 1, pp. 105–119, 2010.

[20] M. Weber, M. Welling, and P. Perona, “Towards automatic discovery of object categories,” in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2. IEEE, 2000, pp. 101–108.

[21] H. Jégou, M. Douze, C. Schmid, and P. Pérez, “Aggregating local descriptors into a compact image representation,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3304–3311.

[22] F. Porikli, O. Tuzel, and P. Meer, “Covariance tracking using model update based on lie algebra,” in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 728–735.

[23] O. Tuzel, F. Porikli, and P. Meer, “Pedestrian detection via classification on riemannian manifolds,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 10, pp. 1713–1727, 2008.

[24] J. B. Tenenbaum and W. T. Freeman, “Separating style and content,” in *Advances in neural information processing systems*, 1997, pp. 662–668.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[26] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *arXiv preprint arXiv:1512.03385*, 2015.

[27] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, “Semantic segmentation with second-order pooling,” *Computer Vision–ECCV 2012*, pp. 430–443, 2012.

[28] T.-Y. Lin, A. RoyChowdhury, and S. Maji, “Bilinear cnn models for fine-grained visual recognition,” in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[29] T.-Y. Lin and S. Maji, “Visualizing and understanding deep texture representations,” in *CVPR*, 2016, pp. 2791–2799.

[30] C. Ionescu, O. Vantzos, and C. Sminchisescu, “Matrix backpropagation for deep networks with structured layers,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2965–2973.

[31] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell, “Compact bilinear pooling,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 317–326.

[32] P. Kar and H. Karnick, “Random feature maps for dot product kernels,” in *Artificial Intelligence and Statistics*, 2012, pp. 583–591.

[33] N. Pham and R. Pagh, “Fast and scalable polynomial kernels via explicit feature maps,” in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 239–247.

[34] S. Kong and C. Fowlkes, “Low-rank bilinear pooling for fine-grained classification,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[35] T.-Y. Lin and S. Maji, “Improved bilinear pooling with cnns,” in *BMVC*, 2017.

[36] S. Liao and S. Z. Li, “Efficient psd constrained asymmetric metric learning for person re-identification,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3685–3693.

[37] S. Bak, E. Corvee, F. Bremond, and M. Thonnat, “Multiple-shot human re-identification by mean riemannian covariance grid,” in *AVSS*, 2011.

[38] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof, “Person re-identification by descriptive and discriminative classification,” in *Image Analysis*, 2011.

[39] B. Ma, Q. Li, and H. Chang, “Gaussian descriptor based on local features for person re-identification,” in *Asian Conference on Computer Vision*. Springer, 2014, pp. 505–518.

[40] M. Gou, X. Zhang, A. Rates-Borras, S. Asghari-Esfeden, M. Sznaier, and O. Camps, “Person re-identification in appearance impaired scenarios,” in *BMVC*, 2016.

[41] B. Ma, Y. Su, and F. Jurie, “Local descriptors encoded by fisher vectors for person re-identification,” in *ECCV Workshops*, 2012.

[42] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, “Person re-identification by local maximal occurrence representation and metric learning,” in *CVPR*, 2015.

[43] E. Pauwels and J. B. Lasserre, “Sorting out typicality with the inverse moment matrix sos polynomial,” in *Advances in Neural Information Processing Systems*, 2016, pp. 190–198.

[44] M. Lovrić, M. Min-Oo, and E. A. Ruh, "Multivariate normal distributions parametrized as a riemannian symmetric space," *Journal of Multivariate Analysis*, vol. 74, no. 1, pp. 36–48, 2000.

[45] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-euclidean metrics for fast and simple calculus on diffusion tensors," *Magnetic resonance in medicine*, vol. 56, no. 2, pp. 411–421, 2006.

[46] N. Gheissari, T. B. Sebastian, and R. Hartley, "Person reidentification using spatiotemporal appearance," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1528–1535.

[47] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *ECCV*, 2008.

[48] B. Prosser, W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by support vector ranking." in *BMVC*, 2010.

[49] W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by probabilistic relative distance comparison," in *CVPR*, 2011.

[50] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2360–2367.

[51] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino, "Custom pictorial structures for re-identification." in *BMVC*, 2011.

[52] T. Wang, S. Gong, X. Zhu, and S. Wang, "Person re-identification by video ranking," in *ECCV*, 2014.

[53] S. Karanam, Y. Li, and R. J. Radke, "Person re-identification with discriminatively trained viewpoint invariant dictionaries," in *ICCV*, 2015.

[54] K. Liu, B. Ma, W. Zhang, and R. Huang, "A spatio-temporal appearance representation for viceo-based pedestrian re-identification," in *ICCV*, 2015.

[55] J. You, A. Wu, X. Li, and W.-S. Zheng, "Top-push video-based person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[56] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian, "MARS: A video benchmark for large-scale person re-identification," in *ECCV*, 2016.

[57] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *CVPR*, 2013.

[58] C. Liang, B. Huang, R. Hu, C. Zhang, X. Jing, and J. Xiao, "A unsupervised person re-identification method using model based representation and ranking," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 771–774.

[59] E. Kodirov, T. Xiang, and S. Gong, "Dictionary learning with iterative laplacian regularisation for unsupervised person re-identification," in *BMVC*, vol. 3, 2015, p. 8.

[60] E. Kodirov, T. Xiang, Z. Fu, and S. Gong, "Person re-identification by unsupervised l1 graph learning," in *European Conference on Computer Vision*. Springer, 2016, pp. 178–195.

[61] W. Li, R. Zhao, T. Xiao, and X. Wang, "DeepReId: Deep filter pairing neural network for person re-identification," in *CVPR*, 2014.

[62] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *ICCV*, 2015.

[63] E. Ahmed, M. Jones, and T. K. Marks, "An improved deep learning architecture for person re-identification," in *CVPR*, 2015.

[64] R. R. Varior, M. Haloi, and G. Wang, "Gated siamese convolutional neural network architecture for human re-identification," in *European Conference on Computer Vision*. Springer, 2016, pp. 791–808.

[65] R. R. Varior, B. Shuai, J. Lu, D. Xu, and G. Wang, "A siamese long short-term memory architecture for human re-identification," in *European Conference on Computer Vision*. Springer, 2016, pp. 135–153.

[66] J. X. W. G. Chi Su, Shiliang Zhang and Q. Tian, "Deep attributes driven person re-identification," in *European Conference on Computer Vision*. Springer, 2016.

[67] N. McLaughlin, J. Martinez del Rincon, and P. Miller, "Recurrent convolutional network for video-based person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[68] S. Gong, M. Cristani, S. Yan, and C. C. Loy, *Person re-identification*. Springer, 2014, vol. 1.

[69] S. Karanam, M. Gou, Z. Wu, A. Rates-Borras, O. Camps, and R. J. Radke, "A comprehensive evaluation and benchmark for person re-identification: Features, metrics, and datasets," *arXiv preprint arXiv:1605.09653*, 2016.

[70] R. Vezzani, D. Baltieri, and R. Cucchiara, "People reidentification in surveillance and forensics: A survey," *ACM Computing Surveys (CSUR)*, vol. 46, no. 2, p. 29, 2013.

[71] L. Zheng, Y. Yang, and A. G. Hauptmann, "Person re-identification: Past, present and future," *arXiv preprint arXiv:1610.02984*, 2016.

[72] B. Ma, Y. Su, and F. Jurie, "Bicov: a novel image representation for person re-identification and face verification," in *British Machive Vision Conference*, 2012, pp. 11–pages.

[73] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li, "Salient color names for person re-identification," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 536–551.

[74] R. Zhao, W. Ouyang, and X. Wang, "Person re-identification by salience matching," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2528–2535.

[75] M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *CVPR*, 2012.

[76] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Learning to rank in person re-identification with metric ensembles," in *CVPR*, 2015.

[77] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, "Local fisher discriminant analysis for pedestrian re-identification," in *CVPR*, 2013.

[78] F. Xiong, M. Gou, O. Camps, and M. Sznaier, "Person re-identification using kernel-based metric learning methods," in *ECCV*, 2014.

[79] L. Zhang, T. Xiang, and S. Gong, "Learning a discriminative null space for person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[80] Y. Zhang, B. Li, H. Lu, A. Irie, and X. Ruan, "Sample-specific svm learning for person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[81] G. Lisanti, I. Masi, A. D. Bagdanov, and A. Del Bimbo, "Person re-identification by iterative re-weighted sparse ranking," *T-PAMI*, vol. 37, no. 8, pp. 1629–1642, 2015.

[82] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based cnn with improved triplet loss function," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[83] A. Mignon and F. Jurie, "PCCA: A new approach for distance learning from sparse pairwise constraints," in *CVPR*, 2012.

[84] M. Hirzer, P. M. Roth, M. Köstinger, and H. Bischof, "Relaxed pairwise learned metric for person re-identification," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 780–793.

[85] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikäinen, and S. Z. Li, "Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes," in *CVPR*, 2010.

[86] C. C. Loy, T. Xiang, and S. Gong, "Multi-camera activity correlation analysis," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1988–1995.

[87] P. M. Roth, M. Hirzer, M. Koestinger, C. Beleznai, and H. Bischof, "Mahalanobis distance learning for person re-identification," in *Person Re-Identification*. Springer, 2014, pp. 247–267.

[88] M. Moakher and P. G. Batchelor, "Symmetric positive-definite matrices: From geometry to applications and visualization," in *Visualization and Processing of Tensor Fields*. Springer, 2006, pp. 285–298.

[89] A. Cherian, S. Sra, A. Banerjee, and N. Papanikolopoulos, "Jensen-bregman logdet divergence with application to efficient similarity search for covariance matrices," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 9, pp. 2161–2174, 2013.

[90] D. Chen, Z. Yuan, B. Chen, and N. Zheng, "Similarity learning with spatial constraints for person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[91] O. Camps, M. Gou, T. Hebble, S. Karanam, O. Lehmann, Y. Li, R. J. Radke, Z. Wu, and F. Xiong, "From the lab to the real world: Re-identification in an airport camera network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 3, pp. 540–553, 2017.

[92] S. Karanam, M. Gou, Z. Wu, A. Rates-Borras, O. Camps, and R. J. Radke, "A systematic evaluation and benchmark for person re-identification: Features, metrics, and datasets," *arXiv preprint arXiv:1605.09653*, 2016.

[93] W. Schwartz and L. Davis, "Learning Discriminative Appearance-Based Models Using Partial Least Squares," in *Proceedings of the XXII Brazilian Symposium on Computer Graphics and Image Processing*, 2009.

[94] W. Zheng, S. Gong, and T. Xiang, "Associating groups of people," in *Proc. of The 20th British Machine Vision Conference (BMVC)*, 2009.

[95] C. C. Loy, T. Xiang, and S. Gong, "Time-delayed correlation analysis for multi-camera activity understanding," *IJCV*, vol. 90, no. 1, pp. 106–129, 2010.

[96] D. Baltieri, R. Vezzani, and R. Cucchiara, "Sarc3d: a new 3d body model for people tracking and re-identification," in *Proceedings of the 16th International Conference on Image Analysis and Processing*, Ravenna, Italy, Sep. 2011, pp. 197–206.

[97] S. Wang, M. Lewandowski, J. Annesley, and J. Orwell, "Re-identification of pedestrians with variable occlusion and scale," in *ICCV Workshops*, 2011.

[98] N. Martinel and C. Micheloni, "Re-identify people in wide area camera network," in *CVPR Workshops*, 2012.

[99] A. Bialkowski, S. Denman, S. Sridharan, C. Fookes, and P. Lucey, "A database for person re-identification in multi-camera surveillance networks," in *DICTA*, 2012.

[100] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning," in *Asian Conference on Computer Vision*. Springer, 2012, pp. 31–44.

[101] W. Li and X. Wang, "Locally aligned feature transforms across views," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3594–3601.

[102] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *T-PAMI*, vol. 32, no. 9, pp. 1627–1645, 2010.

[103] D. Figueira, M. Taiana, A. Nambiar, J. Nascimento, and A. Bernardino, “The HDA+ data set for research on fully automated re-identification systems,” in *ECCV Workshops*, 2014.

[104] P. Dollár, R. Appel, S. Belongie, and P. Perona, “Fast feature pyramids for object detection,” *T-PAMI*, vol. 36, no. 8, pp. 1532–1545, 2014.

[105] A. Das, A. Chakraborty, and A. K. Roy-Chowdhury, “Consistent re-identification in a camera network,” in *European Conference on Computer Vision*, ser. Lecture Notes in Computer Science, vol. 8690. Springer, 2014, pp. 330–345.

[106] A. Dehghan, S. Modiri Assari, and M. Shah, “Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4091–4099.

[107] Z. Zheng, L. Zheng, and Y. Yang, “Unlabeled samples generated by gan improve the person re-identification baseline in vitro,” *arXiv preprint arXiv:1701.07717*, 2017.

[108] R. Benenson, M. Omran, J. Hosang, and B. Schiele, “Ten years of pedestrian detection, what have we learned?” *arXiv preprint arXiv:1411.4304*, 2014.

[109] Y. Li, Z. Wu, S. Karanam, and R. Radke, “Real-world re-identification in an airport camera network,” in *ICDSC*, 2014.

[110] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *Imaging Understanding Workshop*, 1981.

[111] M. Gou, S. Karanam, W. Liu, O. Camps, and R. J. Radke, “Dukemtmc4reid: A large-scale multi-camera person re-identification dataset,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

[112] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, “Performance measures and a data set for multi-target, multi-camera tracking,” in *European Conference on Computer Vision*. Springer, 2016, pp. 17–35.

[113] R. Zhao, W. Ouyang, and X. Wang, “Learning mid-level filters for person re-identification,” in *CVPR*, 2014.

[114] C. C. Loy, C. Liu, and S. Gong, “Person re-identification by manifold ranking,” in *ICIP*, 2013.

[115] S. Karanam, E. Lam, and R. J. Radke, “Rank persistence: Assessing the temporal performance of real-world person re-identification,” *arXiv preprint arXiv:1706.00553*, 2017.

[116] O. Camps, M. Gou, T. Hebble, S. Karanam, O. Lehmann, Y. Li, R. Radke, Z. Wu, and F. Xiong, “From the lab to the real world: Re-identification in an airport camera network,” *IEEE Transactions on Circuits and Systems for Video Technology (T-CSVT)*, vol. PP, no. 99, 2016.

[117] M. Gou, O. Camps, M. Sznaier *et al.*, “mom: Mean of moments feature for person re-identification.”

[118] R. A. Fisher, “The use of multiple measurements in taxonomic problems,” *Annals of eugenics (AE)*, vol. 7, no. 2, pp. 179–188, 1936.

[119] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, “Graph embedding and extensions: a general framework for dimensionality reduction,” *T-PAMI*, vol. 29, no. 1, pp. 40–51, 2007.

[120] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, “Learning locally-adaptive decision functions for person verification,” in *CVPR*, 2013.

[121] D. H. Foley and J. W. Sammon, “An optimal set of discriminant vectors,” *IEEE Transactions on Computers*, vol. 100, no. 3, pp. 281–289, 1975.

[122] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[123] M. Cimpoi, S. Maji, and A. Vedaldi, “Deep filter banks for texture recognition and segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3828–3836.

[124] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, “Multi-scale orderless pooling of deep convolutional activation features,” in *European conference on computer vision*. Springer, 2014, pp. 392–407.

[125] P. Li, J. Xie, Q. Wang, and W. Zuo, “Is second-order information helpful for large-scale visual recognition?” in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[126] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach, "Multimodal compact bilinear pooling for visual question answering and visual grounding," *arXiv preprint arXiv:1606.01847*, 2016.

[127] Z. Yu, J. Yu, J. Fan, and D. Tao, "Multi-modal factorized bilinear pooling with co-attention learning for visual question answering," *arXiv preprint arXiv:1708.01471*, 2017.

[128] A. RoyChowdhury, T.-Y. Lin, S. Maji, and E. Learned-Miller, "Face identification with bilinear cnns," *arXiv preprint arXiv: 1506.01342*, 2015.

[129] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[130] Y. Cui, F. Zhou, J. Wang, X. Liu, Y. Lin, and S. Belongie, "Kernel pooling for convolutional neural networks," in *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[131] J.-B. Lasserre and E. Pauwels, "Sorting out typicality with the inverse moment matrix sos polynomial," in *Neural Information Processing Systems (NIPS 2016)*, 2016.

[132] M. Sznaier and O. Camps, "Sos-rsc: A sum-of-squares polynomial approach to robustifying subspace clustering algorithms," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[133] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based r-cnns for fine-grained category detection," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[134] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[135] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in neural information processing systems*, 2015, pp. 2017–2025.

[136] X. Liu, J. Wang, S. Wen, E. Ding, and Y. Lin, "Localizing by describing: Attribute-guided attention localization for fine-grained recognition." in *AAAI*, 2017, pp. 4190–4196.

[137] B. Zhao, X. Wu, J. Feng, Q. Peng, and S. Yan, "Diversified visual attention networks for fine-grained object classification," *IEEE Transactions on Multimedia*, vol. 19, no. 6, pp. 1245–1256, 2017.

[138] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," Tech. Rep., 2011.

[139] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," Tech. Rep., 2013.

[140] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2013, pp. 554–561.

[141] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[142] "Bilinear CNNs project," http://vis-www.cs.umass.edu/bcnn/, accessed: 2017-11-14.

[143] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International Conference on Machine Learning*, 2013, pp. 1310–1318.

[144] A. Vedaldi and K. Lenc, "Matconvnet – convolutional neural networks for matlab," in *Proceeding of the ACM Int. Conf. on Multimedia*, 2015.

[145] S. Cai, W. Zuo, and L. Zhang, "Higher-order integration of hierarchical convolutional activations for fine-grained visual categorization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 511–520.

[146] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Int. Conf. on Computer Vision*, 2017.

Appendix A

List of Publications

The following list includes all the papers published or submitted for publication by the author during his graduate studies. Papers denoted by * are directly related to the line of research presented in this dissertation. $^+$ indicates equal contribution.

* Xiong, Fei, *Mengran Gou*, Octavia Camps, and Mario Sznaier. "Person re-identification using kernel-based metric learning methods." In European conference on computer vision, pp. 1-16. Springer, Cham, 2014.

Zhang, Xikang, Yin Wang, *Mengran Gou*, Mario Sznaier, and Octavia Camps. "Efficient temporal sequence comparison and classification using gram matrix embeddings on a riemannian manifold." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4498-4507. 2016.

* *Gou, Mengran*, Xikang Zhang, Angels Rates-Borras, Sadjad Asghari-Esfeden, Mario Sznaier, and Octavia Camps. "Person re-identification in appearance impaired scenarios." In Proceedings of the British Machine Vision Conference. 2016

* Camps, Octavia, *Mengran Gou*, Tom Hebble, Srikrishna Karanam, Oliver Lehmann, Yang Li, Richard J. Radke, Ziyan Wu, and Fei Xiong. "From the lab to the real world: Re-identification in an airport camera network." IEEE Transactions on Circuits and Systems for Video Technology 27, no. 3 (2017): 540-553.

Kose, Kivanc, *Mengran Gou*, Oriol Yelamos, Miguel A. Cordova, Anthony Rossi, Kishwer S. Nehal, Octavia I. Camps, Jennifer G. Dy, Dana H. Brooks, and Milind Rajadhyaksha. "Video-mosaicking of in vivo reflectance confocal microscopy images for noninvasive examination of skin lesion (Conference Presentation)." In Photonics in Dermatology and Plastic Surgery, vol. 10037, p. 100370D. International Society for Optics and Photonics, 2017.

* *Gou, Mengran*, Srikrishna Karanam, Wenqian Liu, Octavia Camps, and Richard J. Radke. “Dukemtmc4reid: A large-scale multi-camera person re-identification dataset.” In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. 2017.

Gou, Mengran$^+$, Kivanc Kose$^+$, Oriol Ylamos, Miguel Cordova, Anthony M. Rossi, Kishwer S. Nehal, Eileen S. Flores et al. “Automated video-mosaicking approach for confocal microscopic imaging in vivo: an approach to address challenges in imaging living tissue and extend field of view.” Scientific reports 7, no. 1 (2017): 10759.

* *Gou, Mengran*, Octavia Camps, Mario Sznaier. “moM: Mean of Moments Feature for Person Re-Identification.” In The IEEE International Conference on Computer Vision (ICCV) Workshops. 2017.

* *Gou, Mengran*, Fei Xiong, Octavia Camps, Mario Sznaier: “MoNet: Moments Embedding Network.” In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.

* Karanam, Srikrishna$^+$, *Mengran Gou*$^+$, Ziyan Wu, Angels Rates-Borras, Octavia Camps, Richard J Radke. “A systematic evaluation and benchmark for person re-identification: Features, metrics, and datasets.” IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), 2018