Learning representations for human re-identification

Varior, Rahul Rama

2017

Varior, R. R. (2017). Learning representations for human re-identification. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/70912>

<https://doi.org/10.32657/10356/70912>

LEARNING REPRESENTATIONS FOR HUMAN RE-IDENTIFICATION

RAHUL RAMA VARIOR

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy

2017

*To my parents, my wife, my sister and my in-laws
who always give me love and support.*

Acknowledgments

First and foremost, I would like to express my sincerest gratitude to my supervisor Professor Gang Wang, who has supported and encouraged me patiently throughout my Ph.D. in Nanyang Technological University (NTU). During these years, I have developed not only professional knowledge but also a keen interest in exploring science. This thesis would not have been possible without his insightful guidance and encouragement. I would like to thank my supervisor Professor Alex Chichung Kot for encouraging and supporting me with my research activities at NTU ROSE lab. I would also like to thank School of Electrical and Electronic Engineering, NTU for awarding me the research scholarship and providing me with excellent environment and facilities for my research.

I would like to thank my colleagues Prof. Dong Xu, Prof. Jiwen Lu, my Thesis Advisory Committee members, Prof Ma Kai-Kuang, Prof Qian Kemao and my team members Mr. Amir Shahroudy, Mr. Bing Shuai, Mr. Ting Liu, Dr. Zhen Zuo, Dr. Bing Wang, Mr. Mrinal Haloi, Mr. Wang Xingxing, Mr. Jason Kuen, Mr. Liu Jun, Mrs. Abrar Hamzeh Saad Abdul Nabi, Ms. Kamila Abdiyeva for their useful advices and discussions on my work. I am also indebted to my friends Mr. Josey Mathew, Mr. Harikrishnan P.V., Dr. Amrith Dhananjayan, Dr. Shreejith Shanker, Mrs. Lallu Shreejith, Mr. Rakesh Varier, Dr. Neethu Robinson, Dr. Nandun Thellamuruge, Mr. Animesh Agarwal, Mr. Jithinraj Dharaneendran, Mr. Libin and many others for their support encouragement during my life in Singapore.

Finally, I would like to dedicate this thesis to my parents, my wife, my sister and my in-laws for their constant love, support and encouragement.

Contents

Acknowledgments	iii
List of Figures	viii
List of Tables	xi
List of Abbreviations	xv
List of Notation	xvii
Abstract	xix
Publications	1
1 Introduction	3
1.1 Challenges	5
1.2 Motivation	6
1.3 Major Contributions	9
1.4 Datasets	11
1.5 Organization of the Thesis	13
2 Related Works	15
2.1 Human Re-Identification	15
2.1.1 Feature Representations for Human Re-Identification	16
2.1.2 Metric Learning Frameworks for Human Re-Identification	21
2.2 Deep Learning	21
2.2.1 Feature Learning Algorithms	21
2.2.2 Feature Learning for Human Re-Identification	29
2.3 Concluding Remarks	32

3	Learning Invariant Color Features	35
3.1	Introduction	36
3.2	Invariant Color Features	38
3.2.1	Proposed Approach	40
3.3	Experiments	48
3.3.1	Synthetic Color Constancy dataset	51
3.3.2	Human Re-Identification Datasets and Results	53
3.4	Analysis	57
3.4.1	Number of Training pixels	58
3.4.2	JLCF vs Designed color spaces	59
3.4.3	JLCF vs Sparse Auto-encoder	59
3.4.4	JLCF vs Independent Learning	60
3.4.5	Proposed JLCF vs JLCF without Invariance	60
3.4.6	Comparison with state-of-the-art methods	60
3.4.7	Zero-shot transfer	61
3.5	Conclusion	62
4	Siamese Long Short-Term Memory Network	65
4.1	Introduction	66
4.2	Proposed Framework	69
4.2.1	Learning Contextual Dependency using LSTM	70
4.2.2	The Proposed Siamese LSTM Architecture	71
4.3	Experiments	75
4.3.1	Datasets and Experimental Settings	75
4.3.2	Parameter Tuning	76
4.4	Analysis	77
4.4.1	Internal Mechanisms of the LSTM cell	77
4.4.2	Data Augmentation	81
4.4.3	Fusion with XQDA	81
4.4.4	State-of-the-art Comparison	82
4.5	Conclusion and Future Works	83

5	Gated Siamese Convolutional Neural Network	93
5.1	Introduction	94
5.2	Proposed Model	97
5.2.1	Model Architecture	97
5.2.2	Training and Optimization	104
5.3	Experiments	105
5.3.1	Datasets and settings	106
5.3.2	Results	106
5.4	Analysis	108
5.4.1	Baseline	108
5.4.2	Gated S-CNN	110
5.5	Conclusion and Future Works	114
6	Conclusions and Future Works	117
6.1	Conclusions	117
6.2	Future Works	119
	Appendix	123
	References	129

List of Figures

1.1	(a) Sample images captured using a network of 6 cameras(From Market-1501 dataset [1]). (b) Image sequence of an identity from a single camera (c) Example re-identification scenario with a query image and the retrieved matches.	4
1.2	Summary of major challenges in human re-identification.	7
2.1	Two examples to illustrate the difference in color clouds for the same person in different views due to appearance change in the second view. Red color points denotes the pixel values sampled from upper body part and blue color points denotes that of the lower body part.	20
2.2	Illustration of Unsupervised Techniques (a) Auto-encoder (b) Sparse Coding	22
2.3	Illustration of a Convolutional Neural Network. Classifier not indicated. .	25
2.4	Recurrent Neural Network Architectures. (a) Vanilla RNN (b) Long Short Term Memory (LSTM) RNN (c) RNN unfolded over time.	26
2.5	Siamese Architecture	30
3.1	Examples from CAVIAR4REID, VIPeR and Person Re-ID 2011 datasets to show the appearance changes due to the environmental variables - Illumination, Shading, Camera view angle.	37
3.2	Patch pairs sampled from View A and view B and the histogram and invariant feature learning based encodings.	38
3.3	Proposed invariant color feature learning framework.	41
3.4	Tuning the parameters - Auto-encoder dimensions, Sparse codes dimensions and the number of pixels.	49

3.5	Sample images shown from the synthetic dataset for color constancy experiments.	51
4.1	An example of the human re-identification scenario comparing the LSTM approach (a) and a baseline approach (b) demonstrated on VIPeR dataset.	67
4.2	Proposed siamese LSTM architecture.	70
4.3	Cross-validation curves: (a) hidden size (n) vs Validation performance on Market-1501 dataset. (b) the margin (m) vs validation performance on the Market-1501 dataset.	86
4.4	Training/Validation curves on the CUHK03 dataset.	87
4.5	Qualitative Results showing the internal gating mechanism of the Siamese LSTM architecture	87
4.6	Example 1 of a test image from VIPeR dataset	88
4.7	The different rows of the query image and the corresponding input gate, forget gate and the output gates for example 1.	88
4.8	Example 2 of a test image from VIPeR dataset.	89
4.9	The input gate values of the query, the Rank 1 retrieved match as well as the Rank 2 retrieved match. See text for details.	89
4.10	More examples of successful matches	90
4.11	Some failed cases	91
5.1	Results obtained using a S-CNN. Red, Blue and Yellow boxes indicate some sample corresponding patches extracted from the images along the same horizontal row.	95
5.2	The proposed architecture which is a modified version of our baseline S-CNN.	99
5.3	Gate Visualization: (a) Query paired with its hard-negative (b) Query paired with its positive. The layer 5 input values as well as the corresponding gate values obtained for both pairs are also shown. Boxes of same color indicates corresponding regions in the images.	111
5.4	Validation Curves: (a) The validation performance vs the initial values for $\mathbf{p}$. (b) The validation performance vs the number of layers the matching gate is inserted between, starting from the last as well as the first layers.	115

List of Tables

3.1	Performance (average precision) based on the pixel-wise comparison on the synthetic color constancy dataset with different queries.	52
3.2	Performance (average precision) based on the histogram based comparison on the synthetic color constancy dataset with different queries.	52
3.3	Performance Comparison of the Jointly Learned Color Features (JLCF) different baselines and photometric invariant color spaces on the VIPeR dataset.	54
3.4	Performance Comparison of the Jointly Learned Color Features (JLCF) different baselines and photometric invariant color spaces on the Person Re-ID 2011 dataset.	55
3.5	Performance Comparison of the Jointly Learned Color Features (JLCF) different baselines and photometric invariant color spaces on the CAVIAR4REID dataset.	56
3.6	Performance Comparison of some other color based signatures developed for human re-identification on the VIPeR dataset.	57
3.7	Performance Comparison of some of the alternatives to the proposed approach. SAE - D refers to the Sparse Auto-Encoder with the final dimension kept as D	58
3.8	Performance comparison with different standard color constancy algorithms on the VIPeR, Person Re-ID 2011 and CAVIAR4REID datasets.	58
3.9	Performance Comparison of JLCF combined with AE and CNN and different state-of-the-art results on the VIPeR dataset.	63
3.10	Performance Comparison of JLCF combined with AE and CNN and different state-of-the-art results on the Person Re-ID 2011 dataset.	64

3.11	Performance Comparison of JLCF combined with AE and CNN and different state-of-the-art results on the CAVIAR4REID dataset.	64
3.12	Zero-shot transfer from VIPeR dataset to Person Re-ID 2011 and CAVIAR4REID datasets.	64
3.13	Performance Comparison of our zero-shot transfer method to a domain transfer re-identification (DTR) algorithm [2].	64
4.1	The CMC and mAP on the Market-1501, CUHK03 and VIPeR datasets.	77
4.2	Performance Comparison of proposed LSTM architecture with and without data augmentation on the CUHK03 dataset.	81
4.3	Performance Comparison of proposed LSTM framework and the fusion of the proposed LSTM framework with LOMO-XQDA algorithm on the CUHK03 dataset.	82
4.4	Performance Comparison of state-of-the-art algorithms for the Market-1501 dataset. Results for [3] and [4] are taken from [1].	82
4.5	Performance Comparison of state-of-the-art algorithms for the CUHK03 dataset.	83
4.6	Performance Comparison of state-of-the-art algorithms using an individual method for the VIPeR dataset.	84
5.1	Proposed Baseline Siamese Convolutional Neural Network architecture. .	97
5.2	Performance Comparison of state-of-the-art algorithms for the Market-1501 dataset.	107
5.3	Performance Comparison of state-of-the-art algorithms for the CUHK03 dataset on the ‘detected’ setting.	108
5.4	Performance Comparison of state-of-the-art algorithms using an individual method for the VIPeR dataset.	109
5.5	Baseline variants.	110
5.6	Different feature summarization strategies.	112
1	Performance Comparison of the proposed LCF with different baseline color feature descriptors on the INRIA Holidays and UKB Dataset.	127

2	Performance Comparison of the LCF with different color based approaches based on the INRIA Holidays Dataset	127
3	Performance Comparison of the LCF with different color based approaches based on the UKB Dataset	128

List of Abbreviation

AE	Auto-Encoder
BoF	Bag of Features
BPTT	Back Propagation Through Time
BTF	Brightness Transfer Function
CCTV	Closed Circuit Television
CN	Color Names
CNN	Convolutional Neural Network
DBN	Deep Belief Network
DML	Deep Metric Learning
DMT	Diagonal Matrix Transform
FPNN	Filter Pairing Neural Networks
HE	Hamming Embedding
HOG	Histogram of Oriented Gradients
KISSME	Keep It Simple and Straight-forward Metric Learning
LBP	Local Binary Patterns
LCC	Local Coordinate Coding
LFDA	Local Fisher Discriminant Analysis
LLC	Locality constrained Linear Coding
LOMO	Local Maximal Occurrence
LSTM	Long Short Term Memory
mDAE	marginalized Denoising Auto-Encoder
MG	Matching Gate
PSD	Positive Semi Definite
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
SC	Sparse Coding
S-CNN	Siamese Convolutional Neural Network
SIFT	Scale Invariant Feature Transform
SILTP	Scale Invariant Local Ternary Pattern
SPM	Spatial Pyramid Matching
SVM	Support Vector Machine
XQDA	Cross-view Quadratic Discriminant Analysis

List of Notation

$*$	Convolution
$(\cdot)^T$	Matrix or vector transpose
$\odot$	Element-wise multiplication
$\sqrt{}$	Square root
$\exp(\cdot)$	Exponential function
$\mathbf{x} \in \mathbb{R}^D$	A D -dimensional vector
$\ \mathbf{x} - \mathbf{y}\ _2$	Euclidian distance between vectors $\mathbf{x}$ and $\mathbf{y}$
$\ \mathbf{x}\ _2$	L2 norm of the vector $\mathbf{x}$
$[\mathbf{x} \ \mathbf{y}]$	Concatenation of the vectors $\mathbf{x}$ and $\mathbf{y}$
$\mathbf{X} \in \mathbb{R}^{D_1 \times D_2}$	A matrix $\mathbf{X}$ of size $D_1 \times D_2$
$\ \mathbf{X}\ _F^2$	Frobenius norm of the matrix $\mathbf{X}$

Abstract

This thesis addresses the problem of Human Re-Identification, the task of associating pedestrians over multiple camera views. Human re-identification particularly an interesting problem due to its applications in visual surveillance. Given a probe image of a subject, the objective is to identify a set of matching images of the same subject from a gallery set which are mostly captured by a different camera. Instead of manually searching through images captured by various cameras, it is desirable to automate the human re-identification as it can save enormous amount of manual labor. However, human re-identification is fundamentally a challenging problem due to cluttered backgrounds, ambiguity in visual appearance, variations in illumination, pose and view-point. The goal of this thesis is to present various feature learning architectures in different perspectives to tackle the aforementioned challenges in human re-identification.

Public places are equipped with several thousands of surveillance cameras capturing videos round the clock. Since no biometric aspects such as fingerprint, GAIT or facial cues are accessible from the surveillance videos, visual appearance is the main cue for re-identifying pedestrians. Intuitively, to distinguish pedestrians from surveillance videos, color features can be an important aspect. However, varying illumination and environmental conditions pose a great challenge as the perceived color of the subject may vary. In existing researches, color features are used as it is, i.e. features are extracted from raw pixel values or weakly corrected pixels. In the first part of the thesis, an invariant color feature learning framework is presented to efficiently map and encode the weakly corrected pixel values in an invariant space where the representations of similar colors are close to each other.

In the second part of the thesis, contextual information is incorporated into the local features. Conventional features are extracted locally and independent of other regions.

However, such features lack the global context of the image. Therefore, it is desirable to incorporate the contextual information to the local features. In order to encode such information, a variant of the Recurrent Neural Network architecture called Long Short-Term Memory (LSTM) cells are used. The sophisticated gating mechanisms inside the LSTM cells has the flexibility to selectively propagate the relevant contextual information to the rest of the network.

To eliminate the need for hand-crafted features, an end-to-end trainable Siamese Convolutional Neural Network (S-CNN) architecture is first proposed. However, in conventional S-CNN architectures, the representations of the images are compared only at the final stage when the feature representations mature. In this setting, the network is at risk of failing to capture and propagate subtle local patterns that can distinguish positive pairs from hard-negative pairs. Therefore, a novel gating mechanism is modeled to selectively boost and propagate such common patterns from the middle layers to the final layers of the network. Extensive experimental evaluation and comparisons with baseline algorithms demonstrate the effectiveness of the proposed feature learning models.

Author's Publications

1. **Rahul Rama Varior**, Mrinal Haloi and Gang Wang, “Gated siamese convolutional neural network architecture for human re-identification”, In Proceedings of the *European Conference on Computer Vision (ECCV)*. 2016.
2. **Rahul Rama Varior**, Bing Shuai, Jiwen Lu, Dong Xu, and Gang Wang, “A siamese long short-term memory architecture for human re-identification”, In Proceedings of the *European Conference on Computer Vision (ECCV)*. 2016.
3. **Rahul Rama Varior**, Gang Wang, Jiwen Lu and Ting Liu, “Learning invariant color features for person re-identification”, *IEEE Transactions on Image Processing (T-IP)*, 2016.
4. **Rahul Rama Varior** and Gang Wang, “A data-driven color feature learning scheme for image retrieval”, In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015 .
5. Ting Liu*, **Rahul Rama Varior*** and Gang Wang, “Visual tracking using learned color features”, In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015 .
6. **Rahul Rama Varior** and Gang Wang, “Hierarchical invariant feature learning with marginalization for person re-identification”, *arXiv preprint arXiv:1511.09150*, 2015.

* indicates equal contributions.

Chapter 1

Introduction

Visual Surveillance is an integral part of national security, with various agencies investing heavily on passive visual recording technologies like Closed Circuit Television (CCTV) at critical locations. However, currently the recorded videos have to be analyzed manually which requires enormous amount of time and manual labor. Hence, tools and technologies to automatically analyze and extract relevant information from the recorded videos can be used to significantly reduce the cost and efforts for processing these surveillance videos. Some of the major directions of research in the visual surveillance domain are pedestrian detection, single/multi camera pedestrian tracking, human re-identification, crowd counting, crowd behavior analysis etc. The task of automatically associating pedestrians from multiple CCTV cameras, also known as Human Re-identification (re-identification) is a key problem which has enormous applications in visual surveillance and will be the focus of this thesis.

Human re-identification has received enormous amount of attention in the computer vision research communities. Fundamentally, the goal is to match images of humans from one surveillance camera to the other. Figure 1.1 (a) shows some example images captured using a network of 6 cameras. Given a query image of a particular subject from one camera, the goal is to retrieve the images of the same subject from the other cameras (Figure 1.1 (c)). This can save enormous amount of manual labor in pedestrian tracking across multiple CCTV cameras, tracking the trajectory of a subject and several other real-time and forensic applications.

Current researches rely on several publicly available datasets captured under restricted settings. The largest dataset used for conducting experiments in this thesis consist of

(a)

(b)

(c)

Figure 1.1: (a) Sample images captured using a network of 6 cameras(From Market-1501 dataset [1]). (b) Image sequence of an identity from a single camera (c) Example re-identification scenario with a query image and the retrieved matches.

images captured from a network of 6 cameras [1]. The images are captured with minimal occlusion and several junk/distractor images are present due to mis-detections from the pedestrian detector. Different taxonomies are used in different research works for classifying the existing human re-identification methods. *Single-shot* recognition refers to a one-to-one matching of a pair of images from two cameras. *Multi-shot* recognition refers to matching two sets of images obtained from two different cameras. The two sets of images are obtained by capturing multiple frames from the motion trajectory of the pedestrians as shown in Figure 1.1 (b). In this thesis, we conduct experiments mostly in the multi-shot setting unless stated otherwise.

1.1 Challenges

Human re-identification is an inherently challenging task due to several reasons. Some of the major aspects are listed below.

1. Intra-class variation: In the typical human re-identification setting, images are captured by different cameras with non-overlapping field of views. Hence the environmental conditions such as background, illumination, viewing angle at one location may not be necessarily the same as those at the location of the other cameras. This will result in a very large intra-class variation as the appearance of the images in the other views may substantially differ from the original appearance. As shown in Figure 1.2 (a) and Figure 1.2 (b), the images in view 1 and view 2 are significantly different from each other.
2. Re-identification in crowded scenes and occlusion: Public places such as railway stations, streets and market environments can be often very crowded. This makes it extremely challenging to extract the pedestrian bounding boxes for a subject without any occlusion. Since state-of-the-art datasets employ automatic pedestrian detectors, this will result in several mis-detections. All the above conditions make the re-identification very challenging in crowded scenes as the full appearance of the subject is unavailable. Figure 1.2 (c) shows some example cases with occlusion.

3. Long-term re-identification: Long-term re-identification can pose challenges in two ways mainly in a real-time scenario. First, tracking a pedestrian from location 1 to location 2 becomes increasingly difficult as the separation in space and time increases. To search a person in real-time, it becomes challenging to decide the search space as well as the time frame to conduct the search. Though exhaustive searching across all camera can be a solution, it becomes tedious considering the scalability of the re-identification algorithms. Second, it poses a greater chance that there may be some changes in the appearance of the pedestrians (i.e. change of clothes, accessories etc.).

This thesis mainly focus on addressing the first challenge. Specifically, the aim is to *learn* representations invariant to illumination, pose and view-point changes. Instead of addressing each of these problems individually, the focus is to develop a holistic approach that can handle the problem of large intra-class variation. The fundamental idea is to learn representations where the *distance* between the similar pairs (i.e. images of subjects belonging to the same identity) is lesser compared to the distance between dissimilar pairs (i.e. images of subjects belonging to different identities).

1.2 Motivation

The goal of this thesis is developing robust and invariant feature learning algorithms for the task of human re-identification. As mentioned above, due to the variations in illumination, background, pose and viewpoint, the intra-class variance is likely to be larger than the inter-class variance [5]. In this section, the main motivation behind the works presented in this thesis is explained.

To address the aforementioned challenges, several feature representations were proposed [3, 4, 6–10]. Among the different features, color based features were proven to be an important aspect to distinguish pedestrians in a re-identification system in [10]. It was empirically determined that color features were devoted nearly 75% of the classifier weights. However, variations in illumination causes the perceived color to be different from the actual colors. Human perceptual system has the ability to approximately determine the actual colors even under varying illumination [11]. However, in most of

Figure 1.2: Summary of major challenges in human re-identification.

the existing re-identification algorithms, color features are used *as it is*, i.e. either histograms or color names descriptors are extracted from raw-pixel values. This results in sub-optimal recognition rates as variations in lighting conditions affect the actual pixel values. This can be circumvented to some extent using existing color-constancy algorithms. However, existing color-constancy algorithms rely on several assumptions about the statistics of the color distribution in images, surfaces and its reflectance properties. To develop invariant representations, we propose a data-driven approach to learn a transformation and an encoding scheme from the pixel values in the approximately corrected space. The central idea is to learn the transformation and encoding schemes to project pixel values belonging to same color to be very close to each other. To test the generalization capabilities of the developed feature learning algorithm, it was also applied to an image retrieval application. Further, the proposed color features were combined with learned texture features to achieve promising results on several human re-identification benchmarks.

Invariant representations can be effective to improve the performance of the re-identification system. However, these features are extracted locally and are independent of the features extracted from the other regions of the image. Thus the extracted features lack the spatial dependencies and loose the global context of the image. Therefore, we explore whether discriminative capability of the features can be enhanced by leveraging the contextual information, i.e. from the features extracted at other regions of the image. But, not all the contextual information is necessary as the raw pixel values and the features extracted directly from them have a lot of noisy information. This may result in detecting hard-negatives and several other false positives at top ranks as existing techniques lack the capability to select discriminative features and propagate the ‘relevant’ features to the rest of the network. Recurrent Neural Networks (RNN) have been proven to be very effective in modeling the dependency between sequential inputs [12–14]. The self feedback connections in the RNN cells enable them to memorize the previously ‘seen’ inputs. A variant of the RNN called the Long Short Term Memory (LSTM) cells has the capability to selectively propagate information by leveraging its internal gating mechanisms. Combining all the above ideas, the image is considered as a sequence of spatial regions and the fundamental idea is to adapt RNN units to a Siamese architecture [15] to

model the spatial correlations and learn the contextual features. In addition to modeling the context, the sophisticated gating mechanisms inside the LSTM cells can be used to regulate the flow of relevant contextual information and enhance the discriminative capability of the local features.

In conventional Siamese networks such as the aforementioned Siamese LSTM architecture, the image representations are compared only at the final level. This means that, for an image, its final layer representation does not encode any information from the input image which is paired with it. Thus the obtained representation for an image is fixed regardless of the image it is paired with. In this setting, the network is at risk of failing to capture the subtle local patterns that are essential to distinguish true positive images from hard negative images. First, a baseline Siamese Convolutional Neural Network (S-CNN) is proposed to effectively address the human re-identification problem. To address the aforementioned limitation of the baseline Siamese CNN architecture, we further propose a Gated S-CNN architecture that can explicitly enhance the intricate patterns that are essential to distinguish the positives from hard negatives by selectively boosting them. This is done at multiple levels in the network architecture, thus making the network more matching-aware. The representations are obtained with the knowledge of each other which helps to boost the re-identification performance.

1.3 Major Contributions

Motivated by the ideas mentioned in Section 1.2, three different works have been proposed to learn robust and invariant image representations for the task of human re-identification. The main contributions of the thesis can be summarized as follows.

- To learn invariant color features, we propose a data-driven feature learning framework that learns to project individual pixels into a space where the encoding is invariant to illumination changes. Different from the existing hand-engineered color constancy approaches that corrects the pixel values, the proposed framework aims at learning a meaningful transformation and encoding scheme from raw pixel values so that pixels belonging to similar colors are close to each other in the encoded space. The main advantage over the existing color constancy algorithms is that there are

no explicit assumptions based on the pixel values distribution in the RGB channels of the images, its surfaces and reflectance properties. Based on an auto-encoder and dictionary learning framework, pixel values are transformed and encoded respectively in a higher dimensional space where the representations of pixels belonging to similar color are closer to each other. To test the effectiveness of the proposed algorithm and the robustness of the representations, an image retrieval experiment is first designed using a synthetic color constancy dataset. Our approach is compared with several classical color constancy approaches and the proposed approach outperforms them in most of the settings. Experiments were also conducted on human re-identification datasets and the proposed approach outperforms all the existing color constancy algorithms. To test the generalization capabilities of the auto-encoder based learning and encoding scheme, it is also applied to a standard image retrieval application.

- To strengthen the discriminative power of the image representations, contextual dependency cues were also incorporated in the feature learning framework. Different from the previous approaches using CNNs, hand-crafted features and metric learning frameworks, the proposed approach aims at modeling the dependencies between the features extracted from different regions of the image. Previous feature extraction mechanisms (both learned and hand-crafted features) do not consider the information from nearby regions, and thus the features are extracted locally and are independent of other regions. By adapting the Long Short Term Memory Recurrent Neural Network units to a siamese architecture (Siamese LSTM), we model the spatial dependencies between each local regions of the image and the features are adapted based on the information from other regions of the image. In addition to modeling the spatial context, the sophisticated gating mechanisms inside the LSTM cell can selectively propagate and block information flow into and out of the cell thus regulating information available at the network output. To test the efficiency of the proposed approach, experiments were conducted on several publicly available human re-identification datasets. Comparisons with the baseline algorithms without LSTM units show that the proposed approach can effectively model contextual information in addition to learning invariant features.

Qualitative visualization of the internal gating mechanisms provide an intuitive understanding about the feature selection mechanisms in the LSTM cell. Comparisons with state-of-the-art algorithms show that the proposed algorithm achieves promising results.

- To model the pairwise relationship between images, we build an end-to-end deep neural network architecture with feature selection mechanism and invariant feature learning. In conventional architectures, the image representation is formed without the knowledge of the image that is paired with it, i.e. the comparison is done only at the final layer when the features mature. In the proposed architecture, feature selection mechanism is incorporated from the middle to the top layers of the networks thereby enabling the network to extract subtle features that can distinguish the true positives from hard-negatives. We introduce a Matching Gate (MG) which is a differentiable gating mechanism that can selectively boost subtle common local patterns by comparing the mid-level features of the input image pairs. The MG can be inserted at multiple layers in the network and facilitates end-to-end training of the S-CNN architecture (Gated S-CNN). Experiments were conducted on large human re-identification datasets and we demonstrate improved performance compared to baseline algorithms as well as state-of-the-art algorithms till date.

1.4 Datasets

The major datasets used for conducting experiments in this thesis are the VIPeR [16] dataset, the Person Re-ID 2011 [17] dataset, the CAVIAR4REID [18] dataset, the CUHK03 [19] dataset and the Market-1501 [1] dataset. The details of each of the datasets, ordered by its size, are given below.

Market-1501: The Market-1501 dataset is currently one of the largest publicly available dataset for human re-identification with 32668 annotated bounding boxes of 1501 subjects captured by 6 cameras. Pedestrian bounding boxes are obtained automatically using Deformable Parts Model (DPM) [20] detectors. Since Deformable Part Model (DPM) detectors are used to detect the pedestrians, several false alarm distractors are also present which makes the dataset a challenging one. Following the convention in [1],

the dataset is split into 751 identities for training and 750 identities for testing as done in [1]. Cumulative Matching Characteristics (CMC) [21] and Mean Average precision (mAP) are the metrics used to report the results on this dataset.

CUHK03: The CUHK03 dataset is a challenging dataset collected in the CUHK campus with 13164 images of 1360 identities from two camera views. Evaluation is usually conducted in two settings, ‘labelled’ with human annotated bounding boxes and ‘detected’ with automatically generated bounding boxes (DPM detector). All the experiments presented in this thesis uses the ‘detected’ bounding boxes as this is closer to the real-world scenario. Following [19], evaluation is conducted over twenty random splits and the average results over all splits are reported. There are 100 identities for testing in each split while the rest of the identities are used for training and validation. For the works presented in this thesis, the CMC and the mAP results are reported.

VIPeR: The VIPeR dataset is another challenging dataset for human re-identification which consist of 632 identities captured from two cameras. For each individual, there is only one image per camera view. A stark change in illumination, pose and environment makes this dataset challenging for evaluating human re-identification algorithms. The dataset is split randomly into equal halves and cross camera search is performed to evaluate the algorithms. The CMC results are reported for this dataset as there is only one gallery image for each query image.

Person Re-ID 2011: The Person Re-ID 2011 dataset [17] consists of images extracted from multiple person trajectories recorded from two different, static surveillance cameras. Images from these cameras contain a sheer difference in illumination, background, camera characteristics and a significant view point change. Multiple images per person are available in each camera view. There are 385 person trajectories from one view, 749 from the other and 200 people appear in both views. For our experiments, images of 100 individuals appearing in both views were used for training the system. In this thesis, the CMC results are reported for this dataset.

CAVIAR4REID: CAVIAR4REID [18] is a person re-id evaluation dataset which was extracted from the well known caviar dataset for evaluation of people tracking and detection algorithms. It is a relatively smaller dataset which contains 72 pedestrian images(50 of them in both camera views and the remaining 22 with one camera only) taken from

a shopping mall in Lisbon. Re-identification in this dataset is challenging due to a large variation in the resolution, illumination, occlusion and pose changes. Following [6], 25 identities are used for both training and testing. The CMC results are reported for this dataset in this thesis.

For the invariant color feature learning framework, the algorithm requires manual annotation of patches from image pairs. Due to the annotation complexity and cost, the evaluation was conducted on the VIPeR, CAVIAR4REID and the Person Re-ID 2011 dataset as they are relatively smaller compared to Market-1501 and the CUHK03 dataset. However, for the Siamese LSTM framework as well as the Gated S-CNN framework, experiments were conducted on larger datasets to prove its effectiveness as larger datasets are more suitable for deep learning approaches compared to smaller datasets.

1.5 Organization of the Thesis

This thesis is organized as follows

- Chapter 2 provides an overview of the existing feature extraction and metric learning approaches as well as several deep learning approaches for human re-identification. We also give a comprehensive overview of the related state-of-the-art algorithms in deep learning, specifically, auto-encoders, convolutional neural networks and recurrent neural networks.
- In Chapter 3, the invariant color feature learning algorithm is presented. The detailed formulation and experimental results are presented in this chapter. Further, details about its application in image retrieval is also presented with some results in the Appendix.
- In Chapter 4, the Siamese Long Short Term Memory architecture is presented with the detailed formulation and experiments. Qualitative visualizations demonstrating the internal mechanisms and results are also provided in this chapter.
- In Chapter 5, a baseline S-CNN architecture is presented first and the Matching Gate is introduced to address its limitations. Details of the integration of MG with S-CNN is also presented along with qualitative visualization and results.

- Chapter 6 provides the concluding remarks of this thesis and discussions on some potential future directions for further exploration.

Chapter 2

Related Works

In this chapter, a survey of related works in human re-identification and deep learning is presented. In Section 2.1, an overview of the related hand-crafted approaches for human re-identification is presented. In Section 2.2, popular related deep learning algorithms are presented and in Section 2.2.2, state-of-the-art deep learning models for human re-identification are presented.

2.1 Human Re-Identification

Matching pedestrians across multiple CCTV cameras also known as human re-identification is a very challenging task which has enormous application in visual surveillance. It has received great amount of interest in the academic and industrial research communities in the past decade. In this section, an overview of the related hand-crafted approaches for human re-identification are presented. Majority of the existing research in human re-identification concentrates on two aspects of the problem.

- Developing a feature representation [3, 4, 6–10, 22–26]
- Learning a distance metric [5, 22, 27–35]

First, we review the major feature engineering techniques for human re-identification and further, we review the major metric learning techniques introduced for human re-identification.

2.1.1 Feature Representations for Human Re-Identification

Human Re-Identification is characterized by several challenges such as varying illumination, pose, view-point variations. This can result in large intra-class variation while inter-class variation might be less; i.e. same person might appear very differently across camera views while different persons may look alike in a different view [5]. To tackle these challenges, robust and invariant feature representations are essential. Majority of the hand-engineered features for human re-identification are based on Color [1, 4, 6, 22–24, 31, 34], texture [4, 7, 31] and interest point detectors [4, 31]. Features such as Color histograms [4, 22, 31, 34], Local Binary Patterns (LBP) [34, 36], Color Names (CN) [1, 24], Scale Invariant Feature Transforms (SIFT) [4, 31, 37], Scale Invariant Local Ternary Pattern (SILTP) [22, 30, 38] are commonly used in human re-identification. Gabor-like edges are extracted on the Hue, Saturation and Value channels for each image in [7]. In [4, 31], local patches on a dense grid are extracted and 128 dimensional SIFT features are computed for each patch of size 10×10 at a stride of 5. LBP histogram features on horizontal stripes are extracted for each image in [34]. Image is divided into horizontal stripes to handle the pose changes across views. In [22, 30], SILTP features are used in conjunction with color histograms. Image is divided into 10×10 blocks at a stride of 5 and SILTP histograms are extracted at two scales. The maximal occurrence of each pattern in a horizontal stripe is computed to address the view-point changes. The resulting histogram features called Local Maximal Occurrence (LOMO) features achieve some invariant to view point changes and demonstrate impressive performance on several benchmark datasets. In most of the above works, color histograms are also extracted in addition to the texture features. Besides the features extracted in RGB channels, RGB-D features have also been used for human re-identification [39–42]. RGB-D sensors enable more effective body part localization as well as foreground/background segmentation compared to RGB sensors [42]. It can also be used to extract anthropometric features such as distance between floor and head, ratio of torso and legs etc from the front and rear poses [41].

Color features have been proven to be very important for human re-identification task [10]. In [10], a large pool of texture and color features were combined into a single similarity function and AdaBoost was used to select the most discriminative features

among them. Experiments were conducted to demonstrate the percentage of classifier weights devoted to each of the different types of features. From the model analysis in [10], overall it was observed that 75% of the classifier weights were dedicated to color based features. This proves that color is an important cue that needs to be given attention to for human re-identification. Intuitively, in low resolution surveillance videos, color can be the most informative feature to distinguish the appearance of two pedestrians as other biometric aspects such as face, fingerprint and GAIT are unavailable. Even though the importance of color based features were proven in [10], color based features were not given much attention until recently. However, developing robust color features is very challenging as several factors may cause the appearance of same subject to vary a lot across different camera views. One of the most important factor that contribute to such varying appearance is the variations in illumination. Since the cameras are placed at different locations, the lighting conditions in one location may vary from the other. This affects the perceived color of the subject to be different, i.e. the pixel values in the image will be different from the actual values. Consequently, the features extracted from the images may not be robust to illumination changes. Humans brains possess the ability to ensure that the perceived color of an object remains relatively constant under varying lighting conditions [11]. This particular ability of the human brain is termed as Color Constancy. Land and McCann proposed the Retinex theory [43] to explain this perceptual effect. The central idea behind retinex algorithm is that, the human visual system functions with three independent cone systems peaking on the long, middle and short wavelengths of the visible spectrum and the images formed by these receptors are compared to generate the color sensation [43]. Since then, several computational models have been proposed to achieve color constancy in images. Below, a brief overview of the major color constancy approaches are reviewed. Further, we discuss the major color constancy algorithms applied for human re-identification.

2.1.1.1 Color Constancy

Major algorithms derived from the Retinex theory are the White-patch algorithm [44], Gray-World algorithm [45], the shades of gray algorithm [46] and gray-edge algorithm [47]. In another category of algorithms, illuminant is estimated by obtaining information

about images such as distribution of reflectances and possible light sources. In one of the pioneering works [48], Forsyth proposed the CRULE and MWEXT algorithms to achieve color constancy in Mondriaan world images (i.e., flat and frontally presented image consisting of numerous colored patches) by estimating the illuminant based on the information obtained from images such as reflectances and possible light sources. Following this work, a number of algorithms were proposed for achieving color constancy by illuminant estimation such as Color by correlation [49], Gamut constrained illuminant estimation [50], Gamut mapping using image derivative structures [51], probabilistic methods [52] and genetic algorithm [53]. For a detailed overview of the color-constancy algorithms derived from the Retinex theory and [48], we refer the reader to [54–56].

All of the aforementioned works and the derived works are based on strong assumptions since the color constancy is an under-constrained problem [48, 56]. For example, in [48], the main assumption is constrained gamuts, i.e., only limited number of image colors can be observed under a specific illuminant. Several other assumptions were based on the distribution of colors that are present in an image (e.g., White-Patch, Gray-World and Gray-Edge). In the Gray-World algorithm [45], the major assumption is that, the average reflectance in a scene is achromatic. In the White-Patch algorithm [44], the main assumption was that, the maximum response in the RGB-channels corresponds to a scene point with perfect reflectance. All these works vary in their assumptions and therefore, no color constancy algorithm can be considered as universal.

A Diagonal Matrix Transform (DMT) is the basis of majority of the works [6, 57–59] on color constancy. But the DMT has been proven to be a suboptimal method for achieving color constancy by West and Brill [60] and D’Zmura and Lennie [61]. To improve the performance of the DMT, spectral sharpening [62] derived for each camera can be incorporated. Zaidi [63] proposed a two parameter affine transformation to generate illumination independent color descriptors. Funt and Lewis [64] observed no improvement in the affine model compared to the DMT with spectral sharpening. However, [65] suggests that a generalized diagonal transformation is sufficient to achieve color constancy. Color Eigenflows [66] provides a different approach by developing a statistical linear model that describes how colors change jointly under typical photic parameter changes. Berwick and Lee [57] proposed a log-chromaticity color space to achieve specular, illumination color and illumination pose invariance.

Color features also gained a lot of interest in object recognition. In one of the earliest works, Swain and Ballard [67] identified that color histograms were stable representations over change in views. Funt and Finlayson [68] used ratios of colors from neighbouring location to achieve some illumination invariance. Gevers and Smeulders [59] analysed different color spaces to achieve invariance to a substantial change in viewpoint, object geometry and illumination. But it was observed that the object recognition accuracy degrades substantially for all of the color spaces with a change in illumination color. The invariance property achieved by them has become the basis of several works in tracking, segmentation [69] and image retrieval [70].

2.1.1.2 Color Constancy for Human Re-Identification

Majority of the existing works [1, 4, 7, 31, 34] use color features *as it is*, i.e. the features are computed from the raw pixel values without any correction. The most common approach to capture color features is through color histograms. Histograms are computed by appropriately assigning the pixel values to a particular bin defined by a specific range of pixel values. Another approach to represent color based features is Color Names [71]. The Salient Color Name based color descriptor was proposed for person re-identification in [24]. Each pixel was represented as a vector of 16 color names and the proposed color based descriptor was combined with features extracted from several other standard color spaces to achieve photometric invariance. The log-chromaticity color space was used in [6] to achieve illumination invariance upto translation. The assumption in [6] is that the shape of the color cloud is sufficiently preserved in the $\log \frac{R}{G}$, $\log \frac{B}{G}$ space. But this assumption is violated in many real world images. The color cloud is formed based on sampled observations from upper body and lower part of the body. Figure 2.1 shows some examples of those observations from the VIPeR [16] dataset. The red color points in the color cloud correspond to the pixel values sampled from the upper part of the body in the *rg* color space and blue color points correspond to that of the lower body part. Relying on the color-clouds can be error prone since in a different view, the upper part of the body may have different colors for the same subject.

In another relatively closer work, Porikli [72] and Javed [73] proposed a Brightness Transfer Function (BTF) to find a transformation that maps the appearance of an object in one camera view to the other. But it should be noted that, in [72] the system

Figure 2.1: Two examples to illustrate the difference in color clouds for the same person in different views due to appearance change in the second view. Red color points denotes the pixel values sampled from upper body part and blue color points denotes that of the lower body part.

has to be trained each time when the illumination changes. The mean BTF approach presented in [73] involves computing the mean of several BTFs learned for each training pair of images. However, mean computation removes some vital color information present in small subsets of training data [74]. In [74], this is circumvented by the accumulation of the brightness values of the whole training set before the BTF computation. [74] adopts normalized cumulative histograms of object brightness values for the BTF computation. Therefore, a pixel level correspondence cannot be achieved. A weighted BTF that combines different BTFs computed on several low-level features was proposed in [75] for human re-identification. In [23], different color models were evaluated for human re-identification and a new color space, $[g1, g2, g3]$ was proposed based on the log-chromaticity color space which was used in [6]. The Salient Color Name based color descriptor was proposed for human re-identification in [24]. Each pixel was represented as a vector of 16 color names and the proposed color based descriptor was combined with several other standard color spaces to achieve photometric invariance. Recently, several works [22, 30] uses retinex algorithm to correct the images initially and extract

histograms from the pixel values to achieve approximate illumination invariance.

2.1.2 Metric Learning Frameworks for Human Re-Identification

Some of the prominent metric learning algorithms proposed for person re-identification are [5, 29, 32, 76, 77] etc. Regularized Local Fisher Discriminant Analysis (LFDA) was proposed for person re-identification in [76]. The objective is to maximize the inter-class separability and to minimize the within class variance. To address the non-linearities in feature space, kernel based dimensionality reduction techniques were proposed in [78]. Support Vector Machine (SVM) based Metric learning was proposed in [5], and the idea is to learn decision boundaries that are adaptive to the data samples. In [34], several kernel-based metric learning methods for person re-identification were evaluated and kernel based LFDA was found to be the best performing algorithm for several re-identification datasets. Recently, a subspace and metric learning method called Cross-view Quadratic Discriminant Analysis (XQDA) was proposed in [22]. The algorithm is an extension of the Bayesian face and Keep It Simple and Straight-forward Metric Learning (KISSME) approaches to cross-view metric learning. Another prominent metric learning algorithm that learns an efficient Positive Semi Definite (PSD) logistic metric learning approach with an asymmetric sample weighting strategy was introduced in [30].

2.2 Deep Learning

2.2.1 Feature Learning Algorithms

Modeling complex distributions and functions have been a bottleneck in machine learning. Recent researches indicate that deep architectures [79–82] can efficiently handle these challenges and have shown that better generalization can be obtained. But the idea of learning multi-layer structures are not new. Several works [83, 84], have proposed this idea in the late 1980s. The intuition behind using large networks is that, to learn a complex function that computes the output from input, automatically learning simple functions with multiple layers of abstraction can be beneficial [85]. Additionally, biological evidences substantiate that in the visual cortex, recognition happens at multiple layers [86]. Deep algorithms are most commonly trained through back-propagation. In

Figure 2.2: Illustration of Unsupervised Techniques (a) Auto-encoder (b) Sparse Coding

the late 1980s, the deep neural networks were found to be quite difficult to train due to several problems related to poor initialization of the parameters, saturating activations, vanishing gradients [87] etc. But successful algorithms have been proposed [85, 88–92] to train such large networks in the past decade. Several heuristics and techniques for back-propagation as well as for initializations for ensuring stability of the network have been introduced over the recent years [80, 82, 92–94].

Feature learning techniques can be broadly classified into supervised and unsupervised techniques. Below, we review some of the most common algorithms in both branches of feature learning techniques.

2.2.1.1 Unsupervised Techniques

Recent researches have shown a growing interest in unsupervised feature learning methods such as auto-encoders (AE) [95, 96], sparse coding (SC) [97] and Deep Belief Nets (DBN) [88] since they can be generalized to a large extent and can be trained without any labeled data. Since no handcrafted feature can be considered as universal, learning relations from data can be advantageous. Figure 2.2 shows an illustration of the two most common unsupervised feature learning technique.

Auto-Encoder: Auto-Encoders are a class of unsupervised feature learning algorithm that aims at learning filters from the data based on its reconstruction error. The data

is first projected into a latent space and then reconstructed by transforming the latent representation to the original space. The objective is to minimize the reconstruction error and thus learning meaningful patterns from the data. Figure 2.2 (a) shows an illustration of the simple auto-encoder where the input is projected to a hidden space and then reconstructed at the output. Several variations of the reconstruction error based auto-encoder algorithms [88, 95, 98] have been proposed for unsupervised feature learning. For data-driven feature learning techniques, in-order to achieve a better generalization, they must be trained with a large amount of data. Artificial augmentation of the data by partial corruption was proposed in [95]. But augmented data creates additional computational burden. To circumvent the complexity due to augmented data, marginalized Denoising Auto-Encoder (**mDAE**) was proposed in [99, 100] which is another variant of the traditional denoising auto-encoder [98]. The key technique is to marginalize out the noise by adding a regularization term and thereby reaping the benefits of training on *infinite* data without explicitly corrupting the original data.

Sparse Coding: Sparse Coding is a class of unsupervised techniques that learns a set of basis functions that can efficiently encode that data. The number of basis functions will be larger than the data dimensions (over-complete) so that the obtained codes are sparse. Figure 2.2 (b) shows an illustration of the sparse coding technique. A patch from an image is reconstructed using a subset of a set of basis functions as indicated. The set of basis functions is also called ‘Dictionary’. The resulting coefficients pertaining to the chosen basis gives the sparse codes. Encoding of local descriptors is an extensive topic of study. The bag of features model (BoF) [101], spatial pyramid matching (SPM) [102] have been pioneering among these algorithms and has achieved state-of-the-art in many benchmark datasets. To obtain good performance, researchers have been using non-linear SVMs of very high complexity. This makes it difficult to apply for large sized datasets. An extension of SPM has been proposed in [97] where they make use of sparse coding to achieve a better quantization of local descriptors and they have shown state-of-the-art results. Following the work, several sparse coding algorithms such as Local Coordinate Coding (LCC) [103] and Locality constrained Linear Coding (LLC) [104] have been proposed stating locality of the dictionary elements is more important than sparsity itself. Sparse coding has also been proven to be effective in several other

computer vision applications such as visual tracking [105] and action recognition [106]. In the proposed work, we make use of a sparse coding approach used in [97].

Deep Belief Network: Deep Belief Network is a class of unsupervised technique which is formulated as a multi-layer generative model. Each layer of the DBN encodes the statistical dependencies among the units in the layer below it. DBNs can be viewed as a composition of Restricted Boltzmann Machines (RBM) stacked one upon the other. RBMs have a single hidden layer and no nodes in the hidden layer are connected to each other. Contrastive Divergence [88] is used to train the RBM framework. DBNs have been widely used in dimensionality reduction [89], feature extraction [107, 108], classification [109, 110] and so on.

Though Auto-encoders and RBM share similar ideas, RBM uses a stochastic approach. It uses stochastic units, mostly binary of a Gaussian distribution and the learning step also consist of multiple steps of Gibb's sampling, i.e. to sample hidden vector given the input vector, reconstruct the visible states and so on. Finally, the objective is to adjust the weights so as to minimize the error of reconstruction. In the sense of reconstruction error, sparse coding also shares the same intuition. However, in sparse coding, the aim is to activate the minimal number of filters that are sufficient to reconstruct the data where as in auto-encoders, all the filters that fit the data are activated.

2.2.1.2 Supervised Techniques

Among the supervised techniques in machine learning and specifically computer vision, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) have achieved remarkable success in several applications including image/scene classification [14, 79–82, 111], object recognition [79–81], semantic segmentation [12, 112], image/video captioning [113–115] and a lot more.

Convolutional Neural Networks: The winning model of the ImageNet 2012 competition, AlexNet [111], marked the resurgence of Convolutional Neural Network for computer vision applications. Since then, several CNN architectures have been proposed for almost all computer vision applications of which, some are mentioned above. The central idea behind CNN is to extract simple Gabor like edge or lines in the first layer and progressively extract combination of such low-level patterns to capture more abstract

Figure 2.3: Illustration of a Convolutional Neural Network. Classifier not indicated.

patterns towards the middle and higher layers. Figure 2.3 illustrates a CNN framework that takes in an input image and progressively compute the final feature representation by passing it through successive Convolution, Activation and Pooling layers. Below, details of the main layers of a Convolutional Neural Network are given;

- Convolution - In a Convolution layer, the input is convolved with filters that captures certain patterns from the data. In the lower layers, simple filters such as edge/line filters will be convolved with the input. Towards the middle and higher layers, more abstract patterns will be extracted. A fully connected layer is a special case of convolutional layer when the filter size is same as the input feature map size.
- Activation - Activation function is a function that performs an element-wise linear/non-linear/piece-wise linear mathematical operation on the convolved map (response) to transform it to an output. The most common non-linear activation function used to transform the convolved feature map nowadays is the Rectified Linear Unit (ReLU) [93].
- Pooling - Pooling layers are used to summarize the activations or responses of the convolution operation over a region. It also helps to reduce the size of the feature maps, i.e. the effective dimension. It also brings translation invariance to some extend.

Figure 2.4: Recurrent Neural Network Architectures. (a) Vanilla RNN (b) Long Short Term Memory (LSTM) RNN (c) RNN unfolded over time.

Besides these layers, several other layers such as the local response normalization, batch normalization [94], parametric ReLU [93] etc exist which gives better performances and stability to the network. Once the final layer representations are obtained, the features are passed to a loss function which computes the error with respect to the expected value and the network is optimized to minimize this error.

Recurrent Neural Networks: Recurrent Neural Network (RNN) is a type of deep

neural network that has self-recurrent connections. This enables the network to capture the context information in the sequence and retain the internal states. RNNs have achieved remarkable success in several natural language processing [116, 117], acoustic signal processing [118, 119], machine translation [113, 117] and computer vision [12, 13, 120–123] tasks. The fundamental idea behind RNN is that the connections with previous states enables the network to ‘memorize’ information from past inputs and thereby capture the contextual dependency of the sequential data. Due to the difficulty in learning long range sequences (due to the vanishing gradient problem) [87], Long Short Term Memory (LSTM) [124] Networks were introduced and have been successfully applied to several tasks such as sentence embeddings [116], image caption generation [113], scene labeling [120], action recognition [122, 123], video description [125] etc. In addition to capturing the contextual dependency, LSTM can also selectively allow or block the information flow through the network by using its advanced multiplicative interactions in the cell. Several researchers have conducted an empirical evaluation of different RNN architectures and provided intuitive explanations for the internal interactions in these architectures. For more details, we refer the interested readers to [116, 126–128]. In [116, 129], it has been shown that LSTM cells can detect salient keywords relevant to a topic (context) from sentences or speech inputs. In [130], Pyramidal LSTM was proposed to segment brain images.

Figure 2.4 shows the architecture of the vanilla RNN (Figure 2.4 (a)), the LSTM cell (Figure 2.4 (b)) and the recurrent neural networks unfolded over time (Figure 2.4 (c)). The mathematical formulation of the RNN and LSTM are,

RNN: The hidden state at any time step $\mathbf{h}_t \in \mathbb{R}^n$ can be expressed as a non-linear function of the current input $\mathbf{x}_t \in \mathbb{R}^d$ and the previous hidden state $\mathbf{h}_{t-1} \in \mathbb{R}^n$. The output $\mathbf{y}_t \in \mathbb{R}^k$ at each time step can be further expressed as a non-linear function of the hidden state h_t at that time step.

$$\mathbf{h}_t = f \left(\mathbf{W} \begin{pmatrix} \mathbf{x}_t \\ \mathbf{h}_{t-1} \end{pmatrix} \right) \quad (2.1)$$

$$\mathbf{y}_t = g(\mathbf{W}_{hy}\mathbf{h}_t) \quad (2.2)$$

LSTM: Similar to RNN, the hidden representation $\mathbf{h}_t \in \mathbb{R}^n$ obtained at each step, Eq. (2.5), is a function of the input at the current time step ($\mathbf{x}_t \in \mathbb{R}^d$) and the hidden state at the previous time step ($\mathbf{h}_{t-1} \in \mathbb{R}^n$).

$$\begin{pmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \mathbf{g}_t \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \text{tanh} \end{pmatrix} \mathbf{W}_L \begin{pmatrix} \mathbf{x}_t \\ \mathbf{h}_{t-1} \end{pmatrix} \quad (2.3)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t \quad (2.4)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \quad (2.5)$$

$\mathbf{W}_L \in \mathbb{R}^{4n \times (d+n)}$ denotes the LSTM weight matrix. Sigmoid (*sigm*) and hyperbolic tangent (*tanh*) are the non-linear activation functions which are applied element-wise. $\mathbf{c}_t \in \mathbb{R}^n$ denotes the memory state vector at time t . The vectors $\mathbf{i}_t, \mathbf{o}_t, \mathbf{f}_t \in \mathbb{R}^n$ are the *input*, the *output* and the *forget* gates respectively at time t which modulates whether the memory cell is written to, reset or read from. Vector $\mathbf{g}_t \in \mathbb{R}^n$ at time t is added to the memory cell content after being gated by $\mathbf{i}_t$. Thus the hidden state vector $\mathbf{h}_t$ becomes a function of all the inputs $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t\}$ until time t . The gradients of RNN are computed by back-propagation through time (BPTT) [131].

Internal Mechanisms of LSTM: The input gate can allow the input signal to *alter* the memory state or *block* it (Eq. (2.4)). The output gate can allow the memory contents to be *revealed* at the output or *prevent* its effect on other neurons (Eq. (2.5)). Finally, the forget gate can update the memory cell's state by *erasing* or *retaining* the memory cell's previous state (Eq. (2.4)). These powerful multiplicative interactions enable the LSTM network to capture richer contextual information as it goes along the sequence.

Bias terms are omitted in both RNN and LSTM formulations above for brevity. From the formulations, it can be seen that the hidden state vector $\mathbf{h}_t$ becomes a function of all the inputs $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t\}$ until time t . The recurrent connection of the cell in equation 2.4 acts as the constant error carousel. The sophisticated multiplicative interactions enable LSTM cells to be very powerful for memorizing the sequences, learning dependencies between sequential inputs and selecting the ‘relevant’ information for successful application to several tasks.

Recurrent Neural Networks for Computer Vision: Though RNNs have achieved remarkable success in several natural language processing (NLP) [116, 117], acoustic signal processing [118, 119] and machine translation [113, 117], they have also become very popular in computer vision [12–14, 113–115, 120, 121, 123, 125, 132–135] in the recent years. RNNs have been used in [13, 14] to model the spatial dependencies between local regions for scene classification. To learn the pixel dependencies in different parts of the image, quad-directional RNNs have been proposed for semantic segmentation task in [12, 120, 121]. In [123], LSTM sequence learning is extended to spatio-temporal domain for action recognition. Recurrent Neural Networks have also been used to ‘tie’ different layers in the deep feed forward networks [136]. These RNNs encode data from the same location as opposed to different regions from the same feature map.

A different set of applications of RNNs, is to predict or generate the pixels or different parts of an image. In [134, 135] recurrent neural networks are used to generate/analyze images part by part. In these works, RNNs are not used to model the spatial dependency, but to predict the pixel values as well as to predict which location to be attended in the next iteration. RNNs are also used to combine the feature vectors from different frames in a video sequence [125, 133] for classification and recognition tasks. Another application of RNN is in image/video caption generation and description [113–115, 132]. In these works, LSTM-RNNs are used for language modeling with CNN features of the images as inputs. Different from the above works, the central idea in the proposed framework is to selectively propagate relevant contextual information from the pedestrian images. The proposed architecture also models LSTM units as a siamese network.

2.2.2 Feature Learning for Human Re-Identification

Feature learning algorithms have been proposed for human re-identification in [19, 137–142]. Majority of the feature learning architectures for human re-identification are modeled in a siamese fashion [15]. Figure 2.5 shows the typical siamese architecture. Unlike traditional networks, siamese networks are modeled in a pairwise setting, i.e. inputs are taken as pairs as opposed to single inputs in conventional networks. The objective of most of the siamese networks [15, 143] is to learn an embedding such that objects belonging to the same class are closer to each other and the objects belonging to different classes

Figure 2.5: Siamese Architecture

are far apart. In an interesting work in this direction to learn an invariant mapping, the contrastive loss function [143] was proposed. The objective is to separate objects belonging to different classes by a distance defined by a ‘margin’ while keeping the images of objects belonging to same class as close as possible in the embedding space.

The first Siamese CNN (S-CNN) architecture for human re-identification was proposed in [142]. The system (DML) consists of a set of 3 S-CNNs for different regions of the image and the features are combined by using a cosine similarity as the connection function. Finally a binomial deviance is used as the cost function to optimize the network end-to-end. Local body-part based features and the global features were modeled using a Multi-Channel CNN framework in [139]. Deep Filter Pairing Neural Network (FPNN) was introduced in [19] to jointly handle misalignment, photometric and geometric transformations, occlusion and cluttered background. In [138], a cross-input neighborhood difference module was proposed to extract the cross-view relationships of the features and have achieved impressive results in several benchmark datasets. A recent work [140] also attempts to model the cross-view relationships by jointly learning subnetworks to extract the single image as well as the cross image representations. In [141], domain guided dropout was introduced for selecting the appropriate neuron for the images belonging to a given domain. Different from the above frameworks, the siamese LSTM based framework in Chapter 4 and the CNN based framework in 5 concentrates on modeling

the spatial context as well as learning subtle similarities to discriminate positives from hard-negatives respectively. Both frameworks make use of gating mechanisms for feature selection. Below, a brief review of the different gating mechanisms in deep networks are given.

Gating Mechanisms in Deep Neural Networks: Gating functions have been proven to be an important component in deep neural networks [82, 124]. Gating mechanisms such as the input gates and output gates were proposed in Long-Short Term Memory (LSTM) [124] cells for regulating the information flow through the network. Further, LSTM unit with forget gate [144] was proposed which has the flexibility to reset the internal states based on the inputs. Inspired by the LSTM, Highway Networks [82] were proposed to train very deep neural networks by introducing gating functions into the CNN architecture. More recently, ‘Trust Gates’ were introduced in [123] to handle the noise and occlusion in 3D skeleton data for action recognition. In the Siamese LSTM architecture described in Chapter 4, the gating mechanisms inside the LSTM units help to regulate information flow through the network. The matching gate proposed in Chapter 5 is modeled entirely in a different context in terms of its architecture and purpose; i.e., the goal of the matching gate is to compare the local feature similarities of input pairs from the mid-level through the higher layers and weigh the common local patterns based on the similarity scores. This will enable the lower layers of the network to learn filters that can discriminate the local patterns of positive pairs from negative pairs.

The matching gate architecture described in Chapter 5 can be considered to be very similar to extracting the saliency information from pedestrian images. Some of the major saliency learning approaches for human re-identification are [4, 31, 137, 145]. An exhaustive adjacency constrained search is employed in [4, 31] to compute the patch correspondences and to compute the saliency map. In [137], adjacency constrained patch matching is utilized to compute the discriminative and generalization power and further mid-level filters are learned from the patch clusters. In [145], an unsupervised Generative Topic Saliency model is proposed for localized saliency selection in person images. Different from the above approaches, the proposed matching gate architecture is modeled as a component in an end-to-end trainable deep S-CNN architecture. The matching gate can also be treated as a neuron selection approach which is closely related to the Domain

Guided Dropout model [141]. In [141], a CNN architecture based on the GoogleNet is proposed to jointly train on multiple domains (re-identification datasets) and the neurons corresponding to each domain is selectively activated. However, the proposed matching gate aims to model the similarity between input pairs and selectively boost subtle locally shared information to distinguish hard-negatives from positives.

2.3 Concluding Remarks

In this chapter, a brief overview of the related works in human re-identification as well as common feature learning algorithms that are the motivations for this thesis are presented. In addition to feature learning techniques, a brief overview of the color constancy algorithms is also covered to address the problem of varying illumination settings, which is a fundamental problem in human re-identification.

In the first work presented in this thesis, the variations in illumination which is a fundamental problem in human re-identification is identified and studied. A detailed evaluation of several color constancy algorithms is also presented in Chapter 3 by applying it to a synthetic color constancy dataset as well as human re-identification datasets. An invariant color feature learning algorithm is further proposed to address the problem of illumination variations. Different from the hand-crafted approaches for color constancy, the proposed feature learning algorithm learns a mapping and encoding scheme from the data and is independent of the several assumptions about the statistics of the color distributions that the conventional algorithms follow.

In the second work, a siamese architecture is proposed for human re-identification. However, different from the conventional siamese architecture based on CNN, the proposed architecture is based on Recurrent Neural Networks, specifically, the LSTM cells. The objective is to model the contextual dependencies between the regions of the image. Additionally, the LSTM is also capable of selecting the relevant contextual information which is essential for the network to identify and propagate relevant features to the rest of the network.

In the third work, a baseline siamese CNN architecture is first proposed to address the human re-identification problem eliminating the need for hand-crafted features. The

proposed baseline architecture outperforms most of the existing siamese architecture for human re-identification. Additionally, a novel matching gate is also proposed for feature selection. The matching gate can identify common patterns from the early layers of the network and boost them as the information goes along the further layers of the network. Different from the gating mechanisms inside LSTM units, the proposed gating mechanism compares pairs of inputs and identify the common patterns. Even though the objective is to selectively propagate features, the proposed matching gate is modeled in a different context and in a different application.

Chapter 3

Learning Invariant Color Features

Human Re-Identification is fundamentally a challenging problem due to the change in visual appearance caused by varying lighting conditions, pose and view-point changes. To successfully match images of a subject captured from different cameras, the representations of these two images must be close to each other in the embedding space. To achieve this objective, learning invariant feature is essential. As the images are obtained from public surveillance cameras, additional biometric cues such as face, fingerprint or GAIT are unavailable. Therefore, the appearance of the subject across camera views is the main basis to make decisions. In a pioneering work done by Gray and Tao [10], color features were proven to be an effective cue for human re-identification. However, due to variations in illumination, the perceived color of the subject appears to be different across camera views. Therefore, invariant color features are essential in-order to achieve an illumination invariant representation. Building on these premises, in this chapter, a framework to learn invariant color features is first proposed with detailed analysis on its robustness and generalization capabilities. Further, the invariant color features were combined with other types of learned texture features to achieve promising results on several human re-identification benchmarks.

3.1 Introduction

Variations in illumination, pose and view-point poses some of the major challenges in human re-identification. Even though several hand-crafted features have been developed to address the aforementioned challenges, each of these features capture different aspects. Therefore, a combination of multiple features is desired to capture richer feature representation. Among these features, color based features have been proven to be an important cue for human re-identification [6]. An interesting insight on the importance of color features was demonstrated in an experiment conducted by Gray and Tao [10]. They used AdaBoost for giving weights to the most discriminative features and observed that, over 75% of the classifier weights were given to color based features. These observations support the fact that color has to be given much more attention than other handcrafted features (i.e., engineered features with selected statistical measures and parameters) based on shape, texture and regions. But due to the illumination variations across the camera views, the perceived color of same parts for a particular person appear to be different in the other views. Figure 3.1 shows some example image pairs captured across two different views. For instance, for the third subject in Figure 3.1, the bright yellow pants in view 1 appears very dark in view 2. Taking these observations into consideration and as validated from our experiments, we suggest that using color features *as it is*, i.e., the RGB, HSV or YUV color histograms will not be adequate to achieve a stable and invariant color representation. Hence we propose a multilayer feature learning framework that learns invariant color features from raw pixel values as opposed to histogram or other color based handcrafted features.

The proposed framework aims at learning stable color feature representations for images from both camera views by transforming the pixels to an invariant space. Since the images are captured under different unknown lighting conditions, modeling accurate transformations for each of these illumination settings is practically impossible. Existing algorithms to achieve color-constancy rely on strong assumptions about the statistics of color distribution in the image [146]. This means that the pixel values distribution in the RGB color space of an image is examined to determine values correlated with the scene illumination. Hence, these algorithms can project the image only to an approximate color constant space. Further, it should be noted that, no prior information regarding

Figure 3.1: Examples from CAVIAR4REID, VIPeR and Person Re-ID 2011 datasets to show the appearance changes due to the environmental variables - Illumination, Shading, Camera view angle.

the illumination is given in any of the existing datasets for human re-identification. Therefore, to obtain invariant color features, we propose a feature learning framework to explore the structures and patterns inherent in the image pixels. In the learning step, we take advantage of pairs of pixel values from image pairs and enforce an invariance condition while jointly learning a transformation and encoding scheme. An auto-encoder based framework transforms the 3-dimensional RGB pixel values to a higher dimensional space first and encode them using a dictionary which maps these pixels to an invariant space. These encoded values are pooled over a region and concatenated to form the final representation of an image. This framework can be extended to a multilayer structure for learning complementary features at higher levels.

Detailed experimental evaluation demonstrates the robustness of the proposed approach. From the results, it can be inferred that (1) by learning invariant color features, significant improvement can be achieved in the results when compared with the traditional color histograms and other handcrafted features; and (2) when combined with other types of learned low-level (i.e., the first order information at the pixel level) and high-level features (i.e., more abstract concepts that usually refer to attributes or object classes), it can achieve promising results in several benchmark datasets. To demonstrate the generalization capabilities of the proposed invariant color feature learning framework,

Figure 3.2: Patch pairs sampled from View A and view B and the histogram and invariant feature learning based encodings.

experiments were also conducted using the proposed framework without the invariance criterion by applying it to an image retrieval task (details given in Appendix).

Rest of this chapter is organized as follows. The proposed invariant color feature learning technique along with its mathematical formulation is introduced in Section 3.2. In Section 3.3, we provide the detailed experimental analysis of the proposed approach. Section 3.5 concludes this chapter.

3.2 Invariant Color Features

Hand-engineered features focus on particular cues and add more complexity to the system. Ideally, for each illumination setting, a linear transformation is required to transform it to a canonical illuminant space. Since there is no prior information regarding the illumination for any of the images, estimating accurate linear transformation for all different illumination settings is practically impossible. As it is tedious to compute or learn multiple linear transformations, we adopt one of the prominent illumination correction methods [147] as a preprocessing step to transform the images into a *weak* illumination invariant space. Figure 3.2 shows an example of a pair of images corrected by the aforementioned method and a pair of patches sampled from the corresponding parts of the body. It can be observed that, even in the corrected space, the color histogram (as shown in Figure 3.2 (c)) does not yield an invariant representation.

A robust representation should capture a certain amount of *information* which are the stable structures and patterns in the observed input. Though pixel values are affected by different illumination settings, it can be reasonably assumed that there exists a space where the color patterns are invariant to these variations in illumination. Input

pixels captured under an unknown illuminant will lie away from a color constant space and the objective of color constancy algorithms is to project the pixels back to the color constant space. Therefore as mentioned before, an existing color constancy algorithm is initially used to bring the pixel values *closer* to the color constant space. Since the existing color constancy algorithms are weak correction algorithms, the pixel values require further processing to achieve a robust representation which is stable under varying lighting conditions. Therefore, we propose to learn a transformation or a set of filters to extract stable structures and patterns which bring pixels of corresponding colors to be closer to each other. Additionally, previous works [65, 148] suggest that, to achieve color constancy, linear transformations are sufficient unless camera auto-gain control and transducer non-linearities are taken into consideration. Therefore, we use a linear auto-encoder with an invariance constraint to discover such invariant patterns from the data with the objective that in the transformed space, the representation for pixels of same colors should be as close as possible.

Many researchers have empirically found that encoding schemes for quantizing the local descriptors are essential for good performance. Sparse coding [97, 103] has been found to achieve state-of-the-art performance for many classification problems. In this work, a sparse coding technique is adopted to encode each pixel by a set of dictionary elements or codebooks. It was proven that sparse coding can be used to obtain robust representations that can capture the salient properties of the descriptors [97]. Additionally, biological evidences [149] show the plausibility of sparse coding principle in encoding the sensory inputs in mammalian cortex.

For a consistency between the feature mapping and sparse coding, a joint learning framework [150, 151] was used to obtain a transformation and the codebook simultaneously while enforcing the final encoded representation of each pixel belonging to the same color to be same. As validated from our experiments, we observe that the joint learning framework helps to obtain a robust representation and boost the performance significantly. Sparse auto-encoders with an invariance constraint can be considered as an alternative to the proposed approach. While the proposed approach tries to approximate the mapped features with a set of sparse codes by selecting a few of the dictionary atoms, the deviation of the average activations from a very small value is penalized in sparse

auto-encoders. This introduces certain amount of sparsity to the extracted patterns. But as pointed out in [85], learning multiple layers of relatively simpler functions can learn a better function approximation to map the input to output compared to a single layer mapping. Therefore, the feature learning and encoding scheme is kept separate in the proposed approach instead of enforcing multiple constraints over the single feature mapping scheme. More details are given in the experimental analysis section.

3.2.1 Proposed Approach

As mentioned in the previous sections, the appearance of color changes across camera views due to a stark change in illumination. Figure 3.2 shows an example of pair of patches corrected by a color constancy algorithm which still appears to be of different colors. It can be seen from the Figure 3.2 (c) that the histograms of these corrected patches appear to be very different. Finlayson et al. [62] have shown that the diagonal model is an accurate model to achieve color-constancy for narrow-band (sensitive to single wavelength) imaging sensors. The diagonal model states that the R, G, B values for a pixel under canonical illumination can be obtained by individually scaling the observed R, G and B values of the same pixel under an unknown illumination. But in practice, such narrow band sensors do not exist and the diagonal model is thus considered as an approximate model to correct the images for its illumination changes.

As mentioned in [147], an illumination invariant representation can be obtained by normalizing the image channel-wise with the ℓ_2 norm of the respective channels. Effectively, it cancels the diagonal transformation dependency of the pixels and gives a weak approximation of the canonical illuminant space. Even though this technique does not help to achieve complete illumination invariance, we use ℓ_2 norm as a pre-processing step in our approach. Patches sampled from the images are pre-processed using ℓ_2 norm. We use a linear auto-encoder to transform the pixels into a rich higher dimensional space. These transformed pixel values are encoded using a sparse coding technique to obtain more robust representations. The objective behind the encoding is that the encoded values for corresponding pixels should be the same (or very close). To achieve consistency between the linear transformation and the encoding, we adopt a joint learning strategy to optimize the linear auto-encoder transformation and dictionary learning simultaneously.

The parameters are updated alternatively to find the optimal mapping and encoding for the pixel values. An illustration of our approach is shown in Figure 3.3. The detailed flow of our approach is given in the subsequent sections.

Figure 3.3: Proposed invariant color feature learning framework.

3.2.1.1 Training Patch Collection

To train the system, patch pairs were extracted from the training image pairs manually, since no ground truth on patch correspondence was provided. One of the main constraints with any learning based approach for color constancy is that it is very challenging to create a dataset of real-world images with all possible colors captured under all possible illumination settings and light intensities. Therefore, as a selection principle, we carefully choose patch pairs so that they are distributed among different colors under varying illumination settings i.e. the difference in their colors range from zero (or almost no change in their visual appearance) to a maximum observable change. This is to ensure that no bias is introduced in the sampled dataset at the training stage. From each of the training image pairs in each dataset, we sample 3 pairs of patches. The number of patch pairs were arbitrarily chosen. It is also challenging to collect many patch pairs from each image as in many images, the appearance (clothing) of the pedestrian contains

small texture patterns of multiple colors which is not desired. Sampling multiple patches from the same body part may also result in redundant patch samples. Additionally, the number of samples per image pair can be reduced if the dataset is large. Figure 3.2 shows an example of pair of patches sampled from the VIPeR dataset. Once the patch pairs were extracted, we apply ℓ_2 norm based correction and sample pixel pairs randomly from these patch pairs which is the actual training set for the proposed invariant color feature learning framework. These sampled pixels undergo standard normalization and are used as the input training data for our system. In the experiments section, we have given an analysis on the number of training pixel pairs required.

3.2.1.2 Objective Formulation

Once the training data is prepared, the objective function for learning invariant features is formulated with an invariance constraint for the feature representation obtained from pixel pairs across the two views. The objective is formulated by making use of an auto-encoder and sparse encoding technique. As mentioned in Section 3.2, auto-encoder with invariance constraint is capable of learning filters that can capture stable structures and patterns in the data. At the same time, taking the performance into consideration, a sparse encoding is also applied to the descriptors to represent them compactly. Mathematically:

$$\underset{\mathbf{W}_1, \mathbf{W}_2, \mathbf{D}, \alpha_1, \alpha_2}{\text{minimize}} \quad \ell_{ae} + \ell_{sc} + \varepsilon_{en} + \Omega, \quad (3.1)$$

$$\ell_{ae} = \frac{1}{m} \sum_{v=1}^2 \sum_{i=1}^m \left\| (\mathbf{W}_2^T (\mathbf{W}_1^T \mathbf{x}_i^v + \mathbf{b}_1) + \mathbf{b}_2) - \mathbf{x}_i^v \right\|_2^2 \quad (3.2)$$

$$\ell_{sc} = \frac{\beta}{m} \sum_{v=1}^2 \sum_{i=1}^m \|(\mathbf{W}_1^T \mathbf{x}_i^v + \mathbf{b}_1) - \mathbf{D} \alpha_i^v\|_2^2 \quad (3.3)$$

$$\varepsilon_{en} = \frac{\gamma}{m} \sum_{i=1}^m \|\alpha_i^1 - \alpha_i^2\|_2^2 \quad (3.4)$$

$$\begin{aligned} \Omega &= \lambda (\|\mathbf{W}_1\|_F^2 + \|\mathbf{W}_2\|_F^2) + \rho \|\mathbf{D}\|_F^2 \\ &+ \frac{\eta}{m} \sum_{v=1}^2 \sum_{i=1}^m \left(\sum_{k=1}^d \sqrt{(\alpha_{i(k)}^v)^2 + \delta} \right) \end{aligned} \quad (3.5)$$

where ℓ_{ae} is the auto-encoder loss function, ℓ_{sc} is the loss due to the sparse encoding, ε_{en} is the error of the encoded values and Ω is the regularization term to avoid learning trivial values and to enforce sparsity.

The first term ℓ_{ae} denotes the reconstruction error of each sampled pixel. $\mathbf{x}_i^v \in \mathbb{R}^{3 \times 1}$ denotes the 3 dimensional R,G,B value of the i^{th} pixel from view v . In this chapter, $v = 1$ refers to view A images and $v = 2$ refers to view B images. m is the total number of pixels sampled from each views, i.e. i ranges from 1 to m . $\mathbf{W}_1 \in \mathbb{R}^{3 \times h}$ is the linear transformation matrix that transforms each of the pixels into a h dimensional space and $\mathbf{b}_1 \in \mathbb{R}^{h \times 1}$ is the bias term. Similarly, $\mathbf{W}_2 \in \mathbb{R}^{h \times 3}$ is the transformation of the higher dimensional space into the original 3 dimensional space and $\mathbf{b}_2 \in \mathbb{R}^{3 \times 1}$ is the bias term. Let us denote the collection of all pixels from both views by $\mathbf{X}$. This means, $\mathbf{X} = [\mathbf{X}^1 \quad \mathbf{X}^2] \in \mathbb{R}^{3 \times 2m}$, where $\mathbf{X}^1 = [\mathbf{x}_1^1, \mathbf{x}_2^1, \dots, \mathbf{x}_m^1] \in \mathbb{R}^{3 \times m}$ and $\mathbf{X}^2 = [\mathbf{x}_1^2, \mathbf{x}_2^2, \dots, \mathbf{x}_m^2] \in \mathbb{R}^{3 \times m}$ are the R,G,B values of m randomly sampled pixels from patches extracted from view A images and the corresponding pixels from the patches extracted from view B images respectively. As mentioned in Section 2.1.1.1, previous works [65, 148] have proven that, a linear transformation is sufficient to transform images under an unknown illuminant to images under the canonical illuminant. We borrow this intuition into our work and therefore we use a linear auto-encoder.

The second term, ℓ_{sc} is the encoding term where the linearly transformed pixel values are encoded by a Dictionary. $\mathbf{D} \in \mathbb{R}^{h \times d}$ are the basis vectors (Dictionary or Codebook) to encode each of the transformed pixel values in the h dimensional space where d is the number of such learned basis vectors or dictionary atoms. $\alpha_i^v \in \mathbb{R}^{d \times 1}$ denotes the final encoded sparse representation for the input pixel $\mathbf{x}_i^v$. β denotes the penalty of the sparse

coding loss (ℓ_{sc}). Similar to $\mathbf{X}$, let us denote the sparse codes of $\mathbf{X}^1$ and $\mathbf{X}^2$ by α^1 and α^2 . Therefore, the final encoding, $\alpha = [\alpha^1 \ \alpha^2] \in \mathbb{R}^{d \times 2m}$ is the sparse representation for $\mathbf{X}$.

The third term, ε_{en} denotes the encoding error of the sparse codes of pairs of pixels from the two views. This term enforces the invariance and ensures that the learned filters and encoding scheme are capable of extracting stable and invariant features. It takes ℓ_2 norm of the error of the corresponding pixels from two views. γ is the penalty for the encoding error term (ε_{en}) in the total cost. The final term, Ω ensures that the learned parameters are not trivial. The last term in the right hand side of Ω ensures that the computed codes are sparse. For a simple and straightforward gradient based optimization, we replace $|\alpha_i^v|_1$, the ℓ_1 norm with an approximation that can smooth it at the origin, $\sum_{k=1}^d \sqrt{(\alpha_{i(k)}^v)^2 + \delta}$. Here, $\alpha_{i(k)}^v$ indicates the k^{th} coefficient of the sparse codes α_i^v . Therefore, $|\alpha_i^v|_1$ is approximated by taking the sum of the smooth approximation for all the coefficients of α_i^v . These smooth approximations have been widely used in sparse coding literature [152, 153]. δ is infinitesimally small (1×10^{-4}). λ , ρ and η are the penalties for the respective regularization terms on the right hand side of Ω .

3.2.1.3 Optimization

The optimization is done alternatively between α , $\mathbf{D}$ and $(\mathbf{W}_1, \mathbf{b}_1, \mathbf{W}_2, \mathbf{b}_2)$. We use L-BFGS gradient based optimization procedure to update these values. The gradients with respect to each of the terms are given below.

The loss function given in equation (3.1) is

$$\underset{\mathbf{W}_1, \mathbf{W}_2, \mathbf{D}, \alpha_1, \alpha_2}{\text{minimize}} \quad L \quad (3.6)$$

where

$$L = \ell_{ae} + \ell_{sc} + \varepsilon_{en} + \Omega \quad (3.7)$$

The partial derivative of the loss function w.r.t α_i^v can be given as

$$\begin{aligned}
 \frac{\partial L}{\partial \alpha_i^1} &= -\frac{2\beta}{m} \times \mathbf{D}^T \times (\mathbf{W}_1^T x_i^1 + b_1 - \mathbf{D}\alpha_i^1) \\
 &+ \frac{\eta}{m} \left[\frac{\partial \Omega_{sp}}{\partial \alpha_{i(1)}^1} \quad \cdot \quad \cdot \quad \frac{\partial \Omega_{sp}}{\partial \alpha_{i(k)}^1} \quad \cdot \quad \cdot \quad \frac{\partial \Omega_{sp}}{\partial \alpha_{i(d)}^1} \right]^T \\
 &+ \frac{2\gamma}{m} (\alpha_i^1 - \alpha_i^2)
 \end{aligned} \tag{3.8}$$

$$\begin{aligned}
 \frac{\partial L}{\partial \alpha_i^2} &= -\frac{2\beta}{m} \times \mathbf{D}^T \times (\mathbf{W}_1^T x_i^2 + b_1 - \mathbf{D}\alpha_i^2) \\
 &+ \frac{\eta}{m} \left[\frac{\partial \Omega_{sp}}{\partial \alpha_{i(1)}^2} \quad \cdot \quad \cdot \quad \frac{\partial \Omega_{sp}}{\partial \alpha_{i(k)}^2} \quad \cdot \quad \cdot \quad \frac{\partial \Omega_{sp}}{\partial \alpha_{i(d)}^2} \right]^T \\
 &- \frac{2\gamma}{m} (\alpha_i^1 - \alpha_i^2)
 \end{aligned} \tag{3.9}$$

where $\Omega_{sp} = \left(\sum_{k=1}^d \sqrt{(\alpha_{i(k)}^v)^2 + \delta} \right)$ and [...] indicates the vector containing the derivatives of Ω_{sp} w.r.t each dimension of (α_i^v) . The derivative of Ω_{sp} w.r.t each dimension k , of (α_i^v) can be given as

$$\frac{\partial \Omega_{sp}}{\partial \alpha_{i(k)}^v} = \frac{(\alpha_{i(k)}^v)}{\sqrt{(\alpha_{i(k)}^v)^2 + \delta}} \tag{3.10}$$

The partial derivative of the loss function w.r.t $\mathbf{D}$

$$\begin{aligned}
 \frac{\partial L}{\partial \mathbf{D}} &= -\frac{2\beta}{m} \sum_{v=1}^2 \sum_{i=1}^m (\mathbf{W}_1 x_i^v + b_1 - \mathbf{D}\alpha_i^v) \times (\alpha_i^v)^T \\
 &+ 2\rho \mathbf{D}
 \end{aligned} \tag{3.11}$$

The partial derivative of the loss function w.r.t $\mathbf{W}_1, \mathbf{W}_2, b_1, b_2$

$$\begin{aligned}
 \frac{\partial L}{\partial \mathbf{W}_1} &= \frac{2\beta}{m} \sum_{v=1}^2 \sum_{i=1}^m x_i^v \times ((\mathbf{W}_1^T x_i^v + b_1) - \mathbf{D}\alpha_i^v)^T + \\
 &\frac{2}{m} \sum_{v=1}^2 \sum_{i=1}^m ((\mathbf{W}_2^T (\mathbf{W}_1^T x_i^v + b_1) + b_2) - x_i^v) \times (x_i^v)^T \times \mathbf{W}_2^T \\
 &+ 2\lambda \mathbf{W}_1
 \end{aligned} \tag{3.12}$$

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{W}_2} &= 2\lambda \mathbf{W}_2 + \\ &\frac{2}{m} \sum_{v=1}^2 \sum_{i=1}^m (\mathbf{W}_1^T x_i^v + b_1) \times ((\mathbf{W}_2^T (\mathbf{W}_1^T x_i^v + b_1) + b_2) - x_i^v)^T \end{aligned} \quad (3.13)$$

$$\begin{aligned} \frac{\partial L}{\partial b_1} &= \frac{2}{m} \sum_{v=1}^2 \sum_{i=1}^m \mathbf{W}_2 \times ((\mathbf{W}_2^T (\mathbf{W}_1^T x_i^v + b_1) + b_2) - x_i^v) \\ &+ \frac{2\beta}{m} \sum_{v=1}^2 \sum_{i=1}^m ((\mathbf{W}_1^T x_i^v + b_1) - \mathbf{D} \alpha_i^v) \end{aligned} \quad (3.14)$$

$$\frac{\partial L}{\partial b_2} = \frac{2}{m} \sum_{v=1}^2 \sum_{i=1}^m ((\mathbf{W}_2^T (\mathbf{W}_1^T x_i^v + b_1) + b_2) - x_i^v) \quad (3.15)$$

Initially, α^1 and α^2 are updated based on their gradients while keeping $\mathbf{D}$ and $(\mathbf{W}_1, \mathbf{b}_1, \mathbf{W}_2, \mathbf{b}_2)$ fixed. Then $\mathbf{D}$ is updated keeping α^1, α^2 and $(\mathbf{W}_1, \mathbf{b}_1, \mathbf{W}_2, \mathbf{b}_2)$ fixed. Finally, keeping α^1, α^2 and $\mathbf{D}$ fixed, $(\mathbf{W}_1, \mathbf{b}_1, \mathbf{W}_2, \mathbf{b}_2)$ are updated together.

Theoretically, the gradient based optimization is simple for the above objective functions. But for faster convergence and a good optima, it requires a bit of finesse. With that in consideration, practically, good initializations are required for $\mathbf{W}_1, \mathbf{W}_2$ and $\mathbf{D}$. We give the objective function for the initializations in the next subsection. Initialization helps to achieve faster convergence as well as a better local optima. Joint learning is done until convergence. Equation (3.17) for L_{sc} in Section 3.2.1.4 was minimized using the SPAMS toolbox [154].

During testing, to compute the feature representation for each pixel in an image, we first use $\mathbf{W}_1$ and $\mathbf{b}_1$ to transform it into a higher dimensional space and use the learned dictionary $\mathbf{D}$ to compute the sparse codes. Therefore, the dimensionality of the features for each of the input image will be $M \times N \times d$ where $M \times N \times 3$ is the input image dimensions. Figure 3.2 (d) and (e) show the representation obtained by us using the joint learning framework. It can be visually observed that the error of encoding for the pixel sampled from the two patches is much less than the histogram representation of the patch.

3.2.1.4 Initializations

Initializations for $(\mathbf{W}_1, \mathbf{W}_2, b_1, b_2)$ and $\mathbf{D}$ can be obtained by solving the objective functions L_{ae} and L_{sc} respectively.

L_{ae} can be given as

$$\begin{aligned} \underset{\mathbf{W}, \mathbf{b}}{\text{minimize}} \quad & \frac{1}{m} \sum_{v=1}^2 \sum_{i=1}^m \left\| (\mathbf{W}_2^T (\mathbf{W}_1^T x_i^v + b_1) + b_2) - x_i^v \right\|_2^2 \\ & + \sum_{i=1}^m \left\| (\mathbf{W}_1^T x_i^1 + b_1) - (\mathbf{W}_1^T x_i^2 + b_1) \right\|_2^2 \\ & + \lambda (\|\mathbf{W}_1\|_F^2 + \|\mathbf{W}_2\|_F^2) \end{aligned} \quad (3.16)$$

and L_{sc} can be given as

$$\underset{\mathbf{D}, \alpha}{\text{minimize}} \quad \frac{1}{m} \sum_{v=1}^2 \sum_{i=1}^m \left\| (\mathbf{W}_1^T x_i^v + b_1) - \mathbf{D} \alpha_i^v \right\|_2^2 + \lambda \|\alpha_i^v\|_1 \quad (3.17)$$

3.2.1.5 Multi-layer Framework

The same formulation can be extended to a multilayer framework so that the final representation of the patches sampled from the image pairs are close to each other. To achieve this objective, we encode all the pixels in the sampled patches using the method mentioned in the above section and adopt a max pooling scheme over the 2×2 regions to get the representation of a patch. The *max* pooling strategy makes the representation slightly translation invariant. We further adopt the same formulation in equation (3.1) to obtain the transformation and the dictionary. Once the representation is obtained at the second layer, max pooling is done over 4×4 regions. For simplicity, we keep the dimensions h and d same for the second layer. We observed that further increasing the number of layers did not give any significant advantage over the performance. Also considering the complexity of the approach, we use only two layers for our model.

3.2.1.6 Parameters

The formulation contains several parameters such as the weight penalties of the cost function terms, the dimensions of the linearly transformed space and the number of dictionary

atoms. All the parameters were empirically determined by a 2-fold cross-validation on the training data of VIPeR dataset and kept same for others. The training data was initially split into equal halves and the parameters were varied within a certain range as indicated by the magenta, red and blue ‘dot’ ($\bullet$) marks on the graphs of Figure 3.4. Initially, the number of pixels for training (the amount of training data) was varied and the validation performance is as shown in Figure 3.4(c). Further, we kept the number of training pixel pairs as $100k$ i.e. in our experiments, $m = 100k$. Further, we conducted experiments with the auto-encoder alone by enforcing the invariance constraint. The initialization in the above subsection, equation (3.16) is used to learn the mapping. The dimensions of the new space, h is kept as a parameter. This is analogous to learning a certain number of filters in auto-encoders [95, 155] based on image patches. The experiment was conducted with different values for h . The obtained cross-validation performance is shown in Figure 3.4(a). It can be seen that the performance does not improve much after $h = 60$. Therefore, we fix $h = 60$ for the auto-encoder mapping.

The dimensions of the final sparse codes were also tuned in a similar fashion. We adapted the sparse coding framework with the invariance condition and performed a joint optimization with the auto-encoder mapping. The performance variation at different dimensions is as shown in Figure 3.4(b). It can be seen that at $d = 250$, the performance is at its peak and decreases when we increase the dimensions further. Thus we fixed the sparse code dimension as 250 for all the experiments. The whole framework was then extended to a second layer as mentioned in Section 3.2.1.5 with the same parameters for the auto-encoder and sparse codes dimensions and the final performance is as shown in Tables 3.3, 3.4 and 3.5. The other parameters were also tuned as shown above for the dimensions. The obtained parameters are as follows. $\beta = 1$, $\gamma = 0.1$, $\lambda = 3 \times 10^{-3}$, $\rho = 0.01$ and $\eta = 0.01$.

3.3 Experiments

To validate our algorithm, we first conducted experiments on a synthetic color constancy dataset [156]. Further, experiments were conducted on publicly available human re-identification datasets such as VIPeR [16], Person Re-ID 2011 [17] and CAVIAR4REID

Figure 3.4: Tuning the parameters - Auto-encoder dimensions, Sparse codes dimensions and the number of pixels.

[18]. The characteristics of these three datasets are ideal for the evaluation of the proposed Jointly Learned Color Features(JLCF) since the images were captured from two cameras under varying environments such as indoor and outdoor, bright and dark illumination and different view angles. All the experiments are done without using a mask for removing the background. Below, we list the baseline approaches we compare with.

1. **Hist**: We compare our approach with the 3D histogram generated in RGB, HSV and YUV spaces. The histograms are computed from the images without any pre-processing for illumination changes. We use 6 bins for each of the channels so that the representation is 216 dimensional which is close to the 250 dimensional space proposed in this work. Image is divided into 8×8 blocks with a stride of 4 and for each of those blocks, we compute the histogram. We refer to the histogram representations as RGBHist, HSVHist and YUVHist.
2. **cHist**: cHist corresponds to the histogram of the images in the weakly corrected (ℓ_2 norm based correction) space. As mentioned for **Hist**, the number of bins for each of the channels and the size of the image blocks are kept same. The representation based on the corrected color space is referred to as cRGBHist, cHSVHist and cYUVHist.
3. **rgHist**: *rg*Hist corresponds to the histogram in the *rg* space for the corrected images. *rg* color channels are one of the first photometric invariant color channels proposed. The *rg* space corresponds to

$$r = \frac{R}{R + G + B}, \quad g = \frac{G}{R + G + B} \quad (3.18)$$

The image is divided into blocks of 8×8 and for each of the blocks, we compute histogram of 16 bins for each channel. The final representation for each block will be 256 dimensional.

4. **Opponent:** The opponent color space is invariant to specularly. It can be computed by

$$\begin{bmatrix} O^1 \\ O^2 \\ O^3 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & \frac{-2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.19)$$

Image is divided into 8×8 blocks and the final histogram representation is computed as mentioned in [157].

5. **C:** The C color space adds photometric invariance to shadow shading to the opponent color space. It is computed by normalizing the Opponent descriptor by the intensity.

$$C = \begin{bmatrix} O^1 \\ O^2 \\ O^3 \end{bmatrix}^T \quad (3.20)$$

For 8×8 blocks, the histogram is computed as mentioned above for the opponent color space.

6. **Independent Learning:** The learning strategy will be optimizing the auto-encoder transformation first and then obtaining the sparse codes of the transformed pixel values without joint learning. The objective given in appendix A for initialization is used to find the optimal auto-encoder transformation. After obtaining the transformation, dictionary is learned using ℓ_{sc} in equation 3.1.

7. **JLCF without Invariance(JLCF WI):** To show that the learning invariant features are important, we develop the representation without the encoding error term, i.e., excluding the $\|\alpha_i^1 - \alpha_i^2\|_2^2$ term in equation (3.1).

The final representation of an image in each of the color space is obtained by concatenating the histogram of each blocks in the image.

Figure 3.5: Sample images shown from the synthetic dataset for color constancy experiments.

3.3.1 Synthetic Color Constancy dataset

The efficiency of the proposed feature representation was tested on a synthetic color constancy dataset which was used in [156]. The dataset contains 7 sets of images of indoor objects captured in a laboratory setting under varying illuminations and its original image. For more details on the settings under which the images were captured, we refer the reader to [156]. Some example images are shown in Figure 3.5. To compare the algorithms, we formulate a retrieval problem by using the 7 original images as queries and the mean average precision (mAP) over the entire set of queries is reported as the performance measure. We adopted two different methods to compute the matching score.

1. Pixelwise Comparison: The images were corrected using different color constancy algorithms and the total pixel wise ℓ_1 distance with respect to the query was computed. This distance value is sorted to find the top ranking matches. For the proposed method, only the first layer features were used for the comparison. All the pixels were transformed as well as encoded by using the transformation and dictionary learned from the patches sampled from VIPeR dataset. The same patches were used for computing the color model for color eigenflows as well.

Table 3.1: Performance (average precision) based on the pixel-wise comparison on the synthetic color constancy dataset with different queries.

Algorithm	Raw RGB	rg	C	Opp.	CN	GW	max RGB	SG	GE	ℓ_2 norm	CEF	JLCF	JLCF No ℓ_2	SAE-60	SAE -250
Query 1	0.499	0.684	0.273	0.185	0.192	0.383	0.473	0.480	0.283	0.413	0.436	0.451	0.453	0.445	0.442
Query 2	0.037	0.042	0.028	0.032	0.048	0.033	0.041	0.032	0.042	0.034	0.198	0.258	0.183	0.079	0.079
Query 3	0.169	0.204	0.193	0.240	0.229	0.183	0.162	0.175	0.134	0.182	0.265	0.365	0.344	0.241	0.241
Query 4	0.125	0.128	0.135	0.216	0.209	0.130	0.126	0.125	0.147	0.128	0.160	0.217	0.206	0.128	0.127
Query 5	0.192	0.105	0.189	0.395	0.118	0.277	0.204	0.284	0.339	0.275	0.192	0.268	0.307	0.199	0.199
Query 6	0.221	0.776	0.591	0.252	0.277	0.248	0.257	0.236	0.562	0.263	0.114	0.440	0.438	0.217	0.220
Query 7	0.337	0.135	0.122	0.113	0.442	0.326	0.367	0.312	0.129	0.321	0.307	0.463	0.441	0.240	0.239
mAP	0.226	0.296	0.219	0.205	0.216	0.226	0.233	0.235	0.234	0.231	0.239	0.352	0.339	0.221	0.221

Table 3.2: Performance (average precision) based on the histogram based comparison on the synthetic color constancy dataset with different queries.

Algorithm	Raw RGB	rg	C	Opp.	CN	GW	max RGB	SG	GE	ℓ_2 norm	JLCF	JLCF No ℓ_2	SAE-60	SAE -250
Query 1	0.124	0.170	0.241	0.150	0.278	0.137	0.133	0.138	0.438	0.136	0.553	0.513	0.509	0.477
Query 2	0.049	0.037	0.028	0.034	0.035	0.048	0.042	0.043	0.034	0.046	0.150	0.112	0.036	0.033
Query 3	0.126	0.210	0.185	0.159	0.168	0.121	0.125	0.122	0.125	0.121	0.428	0.296	0.253	0.252
Query 4	0.148	0.304	0.117	0.165	0.198	0.157	0.137	0.144	0.134	0.153	0.369	0.231	0.178	0.165
Query 5	0.479	0.336	0.208	0.379	0.240	0.413	0.418	0.482	0.505	0.453	0.317	0.175	0.243	0.253
Query 6	0.716	0.230	0.497	0.468	0.224	0.698	0.716	0.712	0.614	0.727	0.528	0.381	0.302	0.307
Query 7	0.108	0.255	0.141	0.141	0.208	0.115	0.108	0.120	0.097	0.121	0.445	0.869	0.430	0.394
mAP	0.250	0.220	0.203	0.214	0.193	0.241	0.240	0.252	0.278	0.251	0.399	0.368	0.279	0.269

2. Histogram Comparison: This is similar to the above except that the ℓ_1 distance is computed for the histograms of the corrected images. For the proposed method, we do a max-pooling over 8×8 regions on the features obtained for pixel-wise comparison. We do not compare the color eigenflows model here as the “flow” of one image to another was based on the pixel-wise error between the original and the reconstructed images.

The results are shown in Tables 3.1 and 3.2. The abbreviations used in Tables 3.1 and 3.2 for different algorithms are as follows - (C - C Color space, Opp. - Opponent color space, CN - Color Names, GW - Grey World correction, SG - Shades of Grey Correction, GE - Grey Edge Correction, ℓ_2 - Norm - Correction based on the ℓ_2 norm of the pixels, CEF - Color Eigen Flows). From the results, it can be seen that the proposed approach

outperforms the other color-constancy algorithms and designed color spaces in both cases (for 4/7 cases in pixel-wise comparison and 5/7 cases for histogram based comparison). The mean average precision for the proposed method is the best among all the methods for both the methods. It should be noted that, the proposed approach without any initial correction is also capable of extracting stable features as indicated by the performance on Table 3.1 and 3.2.

3.3.2 Human Re-Identification Datasets and Results

Experiments were conducted on challenging human re-identification datasets such as the VIPeR [16], Person Re-ID 2011 [17] and the CAVIAR4REID [18]. The reader is referred to Section 1.4 for more details on these datasets. First we do the comparison with the baselines approaches. Keeping the settings same for all of the baselines, we use LADF [5] metric learning framework for all the comparisons. For baseline 6, 7 and the proposed JLCF, the matching score of first and second layer is combined to form the final score. The results are reported based on the Cumulative Matching Characteristics(CMC) [21]. For each of the datasets, experimental settings and their evaluations are given in detail in the following subsections.

Even though invariant color features are important, to achieve a good performance, color features alone are insufficient. Therefore, complementary features such as learned low-level and high-level features are added and for comparison with state-of-the-art results. The following learned features are used for fusion with our method.

1. **AE**: Single layer auto-encoder features are learned from patches of size $8 \times 8 \times 3$. 400 filters are learned and the filter response of patches for an image are pooled over 8×8 regions. Vectorizing this representation gives the final feature for a single image. The features learned from a single layer auto-encoder are gabor-like edges.
2. **CNN**: The imagenet pre-trained model of Caffe [158] which follows the architecture in [111] is used to obtain high-level features. The dimensionality of the obtained feature is 4096.

The matching scores obtained from the metric learning framework for each of these features are combined to obtain the final scores for identifying the best match. For each query, we first obtain the gallery matching scores for JLCF and rescale it from 0 to 1. Similarly, we obtain matching scores in the range of 0 to 1 for AE and CNN features. These scores can be summed up to obtain the final matching scores. No weights were used to amplify the *influence* of the matching scores of a particular type of feature.

Table 3.3: Performance Comparison of the Jointly Learned Color Features (JLCF) different baselines and photometric invariant color spaces on the VIPeR dataset.

Method	Rank 1	Rank 5	Rank 10	Rank 15
RGBHist	7.59	27.53	43.35	54.74
HSVHist	11.07	33.23	48.42	60.44
YUVHist	1.90	10.44	18.35	24.05
cRGBHist	7.59	27.53	43.67	54.74
cHSVHist	12.34	38.29	52.53	65.19
cYUVHist	3.48	13.60	22.46	28.16
rgHist	1.90	7.59	16.46	22.78
C Color Space	7.59	26.27	40.50	48.73
Opponent	7.91	25.31	38.29	43.67
Independent Learning	20.25	47.78	64.24	76.89
JLCF WI	19.94	48.73	65.50	75.00
JLCF No ℓ_2	23.42	50.63	64.87	72.78
Proposed JLCF	26.27	51.90	67.09	78.17

3.3.2.1 VIPeR Dataset

For VIPeR, all the images are normalized to a size of 128×48 . We use the same settings as mentioned in [4, 5] for the evaluation and the dataset is split into equal halves which leads to images of 316 individuals for training and testing. Tables 3.3, 3.4 and 3.5 shows the performance comparison of our approach with different baseline methods and photometric invariant color spaces. Experimental results suggest that the encoding

Table 3.4: Performance Comparison of the Jointly Learned Color Features (JLCF) different baselines and photometric invariant color spaces on the Person Re-ID 2011 dataset.

Method	Rank 1	Rank 10	Rank 20	Rank 50
RGBHist	0.8	12.3	25.2	43.7
HSVHist	0.1	7.0	9.9	20.7
YUVHist	0.9	6.6	13.3	28.7
cRGBHist	3.0	13.2	24.5	46.0
cHSVHist	1.0	9.2	17.9	34.0
cYUVHist	1.5	7.1	16.4	30.0
rgHist	0.8	3.6	6.6	17.2
C Color Space	8.1	32.1	40.4	58.9
Opponent	2.8	23.4	33.6	52.6
Independent Learning	8.4	39.3	52.3	67.0
JLCF WI	11.0	39.4	54.1	69.2
Proposed JLCF	17.8	48.1	60.1	70.9

based on the joint learning helps to achieve good performance. Tables 3.3, 3.4 and 3.5 also shows our comparison with the independent learning strategy as well as the joint learning framework without the invariance term, ε_{en} in equation 3.1. Additionally, to prove that the proposed work is efficient in extracting stable patterns even without the ℓ_2 norm based correction, we added an experiment by learning the transformation and encoding parameters without the initial ℓ_2 norm based pre-processing. Results in Tables 3.3, 3.4 and 3.5 shows that the JLCF without pre-processing (JLCF No ℓ_2) performs much better than any other histogram based approaches in corrected images. This proves that the proposed approach can extract robust feature representations which are still stable, though not as good as the one developed with the ℓ_2 norm preprocessing.

Table 3.6 shows the comparison of the proposed JLCF with other color based features proposed for human re-identification. Same experimental settings used for the other baseline methods were used for conducting this experiment. In [6], histogram intersection was used to compute the distance between two image representations for *Hist (2 Parts)*. Hist (2 Parts) signature is a 10 bin histogram computed over the log-chromaticity color

Table 3.5: Performance Comparison of the Jointly Learned Color Features (JLCF) different baselines and photometric invariant color spaces on the CAVIAR4REID dataset.

Method	Rank 1	Rank 5	Rank 10	Rank 20
RGBHist	19.79	65.47	81.89	98.94
HSVHist	18.73	56.42	76.00	93.68
YUVHist	17.05	51.36	68.84	94.52
cRGBHist	24.21	68.42	84.84	97.47
cHSVHist	20.21	59.36	77.47	94.94
cYUVHist	22.73	53.26	72.42	96.84
rgHist	12.84	43.36	68.84	92.63
C Color Space	22.52	56.63	72.21	94.73
Opponent	25.05	58.10	71.78	90.52
Independent Learning	25.47	66.52	86.526	99.36
JLCF WI	26.68	65.47	82.73	98.31
Proposed JLCF	32.63	67.15	87.57	99.36

space for the upper part and the lower part of the body. In Table 3.6, we also report the matching results by plugging Hist (2 Parts) with LADF metric learning framework for a fair comparison with the proposed method. It can be observed that the proposed JLCF outperforms both the results obtained by the Hist(2 Parts) color signature.

An evaluation of the state of the art algorithms using a single method is given in Table 3.9. It can be seen that our method achieves the best result among the different algorithms. To achieve the state of the art result, we combined the features from [5] in addition to the AE and CNN features so that we can get a richer signature. The combination was done in the same way as done for the AE and CNN features. However, using color as a single cue, we achieve comparable results with several state-of-the-art methods based on multiple handcrafted features.

3.3.2.2 Person Re-ID 2011

For the Person Re-ID 2011 dataset, images of 100 individuals appearing in both views were used for training the system.

Table 3.6: Performance Comparison of some other color based signatures developed for human re-identification on the VIPeR dataset.

Method	Rank 1	Rank 5	Rank 10	Rank 15
Hist(2 Parts) [6] with LADF [5]	6.65	24.37	38.29	46.20
Hist(2 Parts) [6] - TPAMI 2013	15.54	32.18	41.77	47.28
Salient Color Names (RGB Space) [24] - ECCV 2014	20.7	47.2	60.6	68.8
Proposed JLCF	26.27	51.90	67.09	78.17

In our experiments we do a multi-shot re-id with the same settings as mentioned in [159] and compare our results with the baselines as well as the state-of-the-art results. Tables 3.3, 3.4 and 3.5 shows the comparison of our approach with the baselines and photometric invariant color spaces. Table 3.9 shows the comparison of our approach with state-of-the-art results in Person Re-ID 2011 dataset. As shown in the results, our method clearly outperforms the baselines, different photometric invariant color spaces and when combined with AE and CNN features, it outperforms the state-of-the-art results.

3.3.2.3 CAVIAR4REID

The experimental settings for this dataset are kept the same as in [6] which leads to 25 identities (10 images per individual) for training and testing. Tables 3.3, 3.4 and 3.5 shows that the proposed JLCF signatures are performing significantly better than the baseline color features and different photometric invariant color spaces. We also compare with other standard approaches by combining JLCF with AE and CNN features and the results are reported in Table 3.9. Similar to the VIPeR dataset, we observed that lack of an optimal score combining mechanism affects the performance at rank 1. However, it should be noted that we achieve the best results at higher ranks.

3.4 Analysis

In this section, we give the analysis of our approach and compare it with the baseline methods for color features.

Table 3.7: Performance Comparison of some of the alternatives to the proposed approach. SAE - D refers to the Sparse Auto-Encoder with the final dimension kept as D .

Method	Rank 1	Rank 5	Rank 10	Rank 15
SAE - 60	10.13	28.80	40.19	50.95
SAE - 250	8.86	25.63	39.56	49.68
JLCF (First Layer only)	22.78	49.37	65.82	77.21

Table 3.8: Performance comparison with different standard color constancy algorithms on the VIPeR, Person Re-ID 2011 and CAVIAR4REID datasets.

Dataset $\Rightarrow$	VIPeR				Person Re-ID 2011				CAVIAR4REID			
Method $\Downarrow$	r=1	r=5	r=10	r=15	r=1	r=10	r=20	r=50	r=1	r=5	r=10	r=20
GW - RGB	8.4	25.6	43.2	50.7	2.8	12.4	24.3	46.1	25.2	64.8	83.2	98.2
maxRGB - RGB	8.4	26.3	42.4	50.3	2.7	13.1	24.5	44.3	26.4	63.4	83.7	98.4
SG- RGB	7.9	26.3	41.2	51.6	3.1	13.7	25.6	46.8	26.7	64.2	83.9	97.1
GE- RGB	8.1	27.2	40.9	52.3	3.1	12.9	26.1	46.0	27.1	63.7	84.4	98.2
ℓ_2 - RGB	7.6	27.5	43.7	54.7	3.0	13.2	24.5	46.0	24.2	68.4	84.8	97.5
GW - HSV	13.4	39.4	52.6	63.3	3.8	10.7	18.1	34.6	22.2	49.2	66.2	92.2
maxRGB - HSV	12.8	36.9	53.1	61.7	0.6	4.8	9.2	20.8	16.2	43.6	65.2	91.4
SG- HSV	11.3	40.1	53.7	62.1	1.2	11.3	17.5	36.2	16.4	45.5	68.4	89.5
GE- HSV	11.7	30.2	39.6	49.2	0.8	5.3	9.2	18.5	18.7	44.3	59.9	92.2
ℓ_2 - HSV	12.3	38.3	52.5	65.2	1	9.2	17.9	34.0	20.2	59.4	77.5	95.5
GW - YUV	2.6	11.3	22.1	26.9	0.6	8.2	11.2	28.6	17.3	52.3	71.2	92.9
maxRGB - YUV	2.2	8.7	15.4	23.1	3.0	8.5	12.9	28.2	16.7	51.2	72.6	95.4
SG- YUV	3.2	12.4	19.1	21.7	1.4	7.9	12.7	33.2	17.2	49.7	71.1	96.1
GE- YUV	3.4	9.2	15.1	25.1	3.6	12.6	18.2	34.2	20.9	46.8	69.7	93.2
ℓ_2 - YUV	3.5	13.6	22.5	28.2	1.5	7.1	16.4	30.0	22.7	53.3	72.4	97.0
JLCF	26.3	51.9	67.1	78.2	17.8	48.1	60.1	70.9	32.6	67.2	87.6	99.4

3.4.1 Number of Training pixels

Since the proposed algorithm is a feature learning based model, the amount of training data (specifically, number of pixel pairs sampled) affects the performance of the system. Figure 3.4(c) shows the variation of rank 1 performance of the system with respect to the number of pixel pairs sampled. It can be seen that the performance does not improve significantly when the number of pixel pairs sampled is more than 10^5 . The collected training patches were distributed across different colors sampled under varying illumination. Therefore, the sampled pixel pairs should be uniformly distributed among them to avoid any bias to a particular color or a specific illumination setting. As the

number of training pixels are reduced, the amount of data corresponding to different illumination settings are reduced and the resulting learned model becomes inferior.

3.4.2 JLCF vs Designed color spaces

Color histograms are representations which capture the color distribution. However, the difference in illumination can cause a significant change in the appearance which is caused by the variation in the RGB pixel values. Therefore, without the illumination correction, the histogram will not be a robust representation for color images. Using ℓ_2 norm, we do a correction for each of the images and then obtain the histogram in such a corrected space, the **cHist**. Our approach uses a weak illumination correction algorithm and learn an optimal transformation to encode the pixel values in such a way that pixels corresponding to same patches are close enough. The other photometric invariant color spaces can be considered as handcrafted features addressing specific cues as mentioned in Section 3.3. Table 3.1, Table 3.2 and Tables 3.3, 3.4 and 3.5 shows the comparisons of our method with the baseline approaches and it can be seen that JLCF clearly outperforms most of the representations for color constancy dataset and all of them in human re-identification. This is due to the fact that the ℓ_2 norm based correction which is inspired from the diagonal model is a weak illumination correction due to the strong assumptions. The comparison also shows that learned invariant color features are better than handcrafted color features. From the results in Table 3.1, Table 3.2 and Tables 3.3, 3.4 and 3.5, it can be seen that even though “**JLCF No ℓ_2** ” is inferior to the performance achieved by using the ℓ_2 norm based correction it is better than the histogram representations for corrected images. This indicates that the representation obtained even without any ℓ_2 norm based pre-processing can still capture the stable patterns in the pixels.

In addition to Tables 3.3, 3.4 and 3.5, we also report some additional results based on the histograms computed on the original images corrected by the standard color-constancy algorithms in Table 3.8.

3.4.3 JLCF vs Sparse Auto-encoder

To prove that the proposed approach is more efficient than the sparse auto-encoder, we conducted experiments with the sparse auto-encoder on the synthetic color constancy

dataset as well as the VIPeR dataset. Results reported in Table 3.1, Table 3.2 and Table 3.7 indicate that the proposed architecture is more efficient than sparse auto-encoders. As explained in Section 3.2, multiple layers of relatively simpler functions can learn better approximations to the intended function compared to a single layer mapping with multiple constraints. For the proposed objective function, even though the auto-encoder and sparse coding framework is in a single layer, they can be considered as two separate functions which are individually pre-trained (initializations) and fine tuned. Therefore, it results in a better local optima and has better generalization capability than the sparse auto encoder framework with invariance and sparsity constraints.

3.4.4 JLCF vs Independent Learning

As mentioned in Section 3.2, to be consistent with each other, the linear auto-encoder transformation and dictionary for sparse coding must be learned jointly. As it can be seen from the results in Tables 3.3, 3.4 and 3.5, the joint learning improves the performance significantly for all the datasets. This is due to the fact that, for the encoding of each pixel, an optimal dictionary which can give same representation for pixels of same color has to be learned together with the linear transformation.

3.4.5 Proposed JLCF vs JLCF without Invariance

To prove that the invariance term in the objective function is necessary to capture the required stable patterns, we conducted experiments with and without the color constancy term (ε_{en}) and report the results in Tables 3.3, 3.4 and 3.5. It can be seen that the invariant encoding improves the performance significantly. This is due to the fact that, without the encoding error term, the objective function merely encodes the pixels in a new space without any correction for the varying lighting conditions. But it should also be noted that the sparse encoding of linearly transformed pixel values results in a much better representation than histograms.

3.4.6 Comparison with state-of-the-art methods

To compare with the state-of-the-art results, the matching scores obtained using JLCF is fused with the matching scores obtained using AE and CNN features. The proposed

JLCF, with color features alone in the RGB color space, outperforms several recent methods such as [4, 31, 137] at Rank 10 and 15 and exhibit comparable performances at lower ranks. All the methods mentioned in Table 3.9 that outperforms the proposed JLCF, works based on an ensemble of complementary color and texture features. It should be noted that even though color features are important for human re-identification, color alone would not be sufficient for a good performance. Therefore, to add complementary information, we combine learned low-level and high-level features to the proposed JLCF and conducted experiments. By a combination of these features, we obtained state-of-the-art results as shown in Table 3.9. The results indicate that the proposed color feature (JLCF) is complementary to these texture features. To differentiate the comparisons between supervised and unsupervised techniques, we have split the results in Table 3.9 (a) into two subtables. The first subtable shows the results of unsupervised methods and the second subtable shows the results of supervised methods.

3.4.7 Zero-shot transfer

To study whether the learned dictionary and transformation matrix can be generalized, experiments were conducted by doing a zero-shot transfer, i.e., by training on one dataset and testing on other datasets without any supervised training or fine-tuning on the target dataset. Specifically, the transformation matrix and the dictionary was trained on VIPeR dataset and testing was done on Person Re-ID 2011 and CAVIAR4REID datasets. Results are reported in Table 3.12. Even though the performances does not match the performances obtained while training on the same dataset, it outperforms the histogram based representation on other specially designed color spaces (comparing Tables 3.3, 3.4 and 3.5 and 3.12). We have also given a comparison of the blind learning VIPeR to Person Re-ID 2011 with a domain transfer re-identification algorithm [2]. In [2], transfer learning is performed by calibration from source domain with minimal annotation on the target domain. It can be observed that the proposed system can outperform [2] without any label information in the target domain. In the proposed system, a set of filters as well as dictionary elements were learned during the training phase and these filters can be generalized. Similar observations can be seen in several learning based methodologies. For example, CNN architectures trained on one dataset can be generalized successfully to other datasets [171].

3.5 Conclusion

In this chapter, we propose a novel framework for learning invariant color features to handle varying illumination settings across two camera views. In contrast to the previous works based on auto-encoders and sparse coding, we combine them to learn a robust encoding jointly by forcing similar colors to be close to each other. We have evaluated the proposed approach on a synthetic color-constancy dataset as well as publicly available human re-identification datasets and demonstrated superior performance compared to several standard color-constancy algorithms as well as specially designed color spaces. By combining with other types of learned low-level and high-level features, we achieve promising results in several benchmark datasets for human re-identification. For testing the generalization capability of the proposed auto-encoder based feature learning and encoding framework, the feature learning algorithm is also tested on an image retrieval problem without the invariance condition (details given in Appendix). Partial work presented in this chapter has been published in [172].

Invariant features provide promising improvements in the re-identification performance. However, the features are extracted locally and independent of other regions. In the next chapter, we explore whether discriminative capabilities of the local features can be improved by leveraging the contextual information (i.e. features from other regions of the image). A variant of the Recurrent Neural Network architecture called the Long Short-Term Memory (LSTM) unit is modeled in a Siamese fashion to learn the contextual dependency of the local features. In addition to modeling the contextual dependencies, we also explore whether the discriminative capabilities of the local features can be further enhanced by the feature selection techniques inside the LSTM cells.

Table 3.9: Performance Comparison of JLCF combined with AE and CNN and different state-of-the-art results on the VIPeR dataset.

Method	Rank 1	Rank 5	Rank 10	Rank 15
Unsupervised Methods				
ELF [10] - ECCV 2008	12.00	41.50	59.50	68.00
SDALF [3] - CVPR 2010	19.87	38.89	49.37	58.46
eBiCov [7] - BMVC 2012	20.66	42.00	56.18	63.11
CPS [18] - BMVC 2011	21.84	44.00	57.21	65.18
PatMatch [4] - CVPR 2013	26.90	47.46	62.34	73.41
Supervised Methods				
WBTF [75] - ICPR 2012	21.99	46.84	59.97	70.00
SalMatch [31] - ICCV 2013	30.15	52.31	65.53	73.41
Mid-level Features [137] - CVPR 2014	29.11	52.34	65.95	73.92
LADF [5] - CVPR 2013	29.43	63.29	76.27	83.23
VWCM [26] - ECCVW 2014	30.70	62.97	75.95	81.01
CMWCE [23] - ICPR 2014	37.6	68.1	81.3	87.4
Salient Color names [24] - ECCV 2014	37.8	68.5	81.2	87.0
Deep ML [142] - ICPR 2014	28.2	59.3	73.5	81.2
LFDA [76] - CVPR 2013	24.1	51.2	67.1	-
QAF [160] - CVPR 2015	30.2	51.6	62.4	-
ConvNet [138] - CVPR 2015	34.8	63.7	75.8	-
LX [22] - CVPR 2015	40.0	68.1	80.5	-
PRCSL [161] - ICCV 2015	34.8	68.7	82.3	-
MLAPG [30] - ICCV 2015	40.7	69.9	82.3	-
MT-LORAE [162] - ICCV 2015	42.3	72.2	81.6	-
Semantic [163] - CVPR 2015	41.6	71.9	86.2	-
DGDropout [141] - CVPR 2016	38.6	-	-	-
SI-CI [140] - CVPR 2016	35.8	67.4	83.5	-
SS-SVM [164] - CVPR 2016	42.7	-	84.3	-
MCP-CNN [139] - CVPR 2016	47.8	74.7	84.8	-
HGD [165] - CVPR 2016	49.7	79.7	88.7	-
DNS [166] - CVPR 2016	51.7	82.1	90.5	-
HER [167] - BMVC 2016	53.0	79.8	89.6	-
SCSP [168] - CVPR 2016	53.5	82.6	91.5	-
Proposed JLCF	26.27	51.90	67.09	78.17
Proposed JLCF + AE + CNN + [5]	38.00	70.25	81.01	87.03

Table 3.10: Performance Comparison of JLCF combined with AE and CNN and different state-of-the-art results on the Person Re-ID 2011 dataset.

Method	Rank 1	Rank 10	Rank 20	Rank 50
Descr. Model [17] - SCIA 2011	4	24	37	56
RPML [159] - ECCV 2012	15	42	54	70
Proposed JLCF	17.8	48.1	60.1	70.9
Proposed JLCF + AE + CNN	21.5	49.0	62.7	73.0

Table 3.11: Performance Comparison of JLCF combined with AE and CNN and different state-of-the-art results on the CAVIAR4REID dataset.

Method	Rank 1	Rank 5	Rank 10	Rank 20
HPE [169] - PRL 2012	9.7	33.2	55.6	76.3
LF [33] - CVPR 2013	36.1	51.2	88.6	97.5
LADF [5] - CVPR 2013	29.64	62.01	78.52	94.23
SSCDL [170] - CVPR 2014	49.1	80.2	93.5	97.9
Proposed JLCF	32.63	67.15	87.57	99.36
Proposed JLCF + AE + CNN	45.89	80.84	94.10	100.00

Table 3.12: Zero-shot transfer from VIPeR dataset to Person Re-ID 2011 and CAVIAR4REID datasets.

Person Re-ID 2011				CAVIAR4REID			
r=1	r=10	r=20	r=50	r=1	r=5	r=10	r=20
9.5	29.6	44.5	60.8	27.79	61.89	82.95	99.16

Table 3.13: Performance Comparison of our zero-shot transfer method to a domain transfer re-identification (DTR) algorithm [2].

Method	Rank 1	Rank 5	Rank 10	Rank 15
DTR [2] - ARTEMIS 2013	2.5	15.8	21.5	28.4
JLCF	9.5	29.6	44.5	60.8

Chapter 4

Siamese Long Short-Term Memory Network

Invariant color and texture features have been proven to be very effective to address illumination variations, pose and view-point changes for human re-identification (refer to Chapter 3). However, all the hand-crafted as well as learned features are extracted locally and independent of other regions. Thus the network fails to adapt to the nature of the image and loses the global context of the image. Therefore, it is desirable to leverage the spatial context (i.e. information from other regions of the image) during feature extraction. However, all the spatial context might be unnecessary as the local features extracted from raw pixel values may contain a lot of noisy information. The existing approaches to human re-identification lack mechanisms to selectively propagate the relevant features to the network. In this chapter, we explore the above two aspects of the feature learning framework, i.e. (1) Modeling the spatial dependency between the different parts of the image and (2) selectively propagating the relevant contextual information to enhance the discriminative capacity of the network.

4.1 Introduction

Invariant features have been shown to be successful to address several challenges including variations in illumination, pose and view-point in human re-identification. Many of the existing research in human re-identification concentrate on developing robust feature representations [22–24, 26]. With the recent advance of deep learning technologies for various computer vision applications, researchers also developed new deep learning architectures [19, 138–142] based on Convolutional Neural Networks (CNNs) for the human re-identification task. But, most of these handcrafted features as well as learned features have certain limitations. When computing histograms or performing convolution followed by max-pooling operation for example, the features are extracted locally and thus are independent of those features extracted from other regions [130]. Thus the network or the system fails to adapt itself to the nature of the image and loses the global context of the image. Therefore, it is desirable to leverage the contextual information (i.e. the information from other regions of the image) during feature extraction/learning. However, not all the spatial dependencies might be relevant for the image under consideration. The features obtained from the raw pixel values contain a lot of redundant or noisy information. This may result in several false alarms, i.e. to detect hard-negatives as true match. Figure 4.1 shows an example of a human re-identification scenario. Figure 4.1 (b) shows a result obtained by a baseline algorithm that learns invariant features. Images in blue boxes represent a query image. The similarities in the background, the color of the query subject’s lower part of the body and the light shades on the upper part of the query image results in several mis-detections. Even though these information are not particularly useful, it still contribute towards the ‘decision making’. Therefore, it is desirable to choose the relevant information from the probe and gallery images. In this chapter, we explore whether the discriminative capability of local features can be enhanced with feature selection techniques by leveraging the contextual information of the image. Figure 4.1 (a) shows an example using our approach. The network is able to retrieve the correct match (shown in green box) from the gallery images. Additionally, the other images retrieved in the top ranks are visually very similar to the query image.

Recurrent Neural Network (RNN) architectures have been developed to successfully model spatial correlations [12–14] and adapt the local representations to extract more

Figure 4.1: An example of the human re-identification scenario comparing the LSTM approach (a) and a baseline approach (b) demonstrated on VIPeR dataset.

discriminative features. The self recurrent connections in RNNs enable them to learn representations based on inputs that it has previously ‘seen’. Thus the features learned by RNNs at any point can encode the spatial dependencies and ‘memorize’ them. Since all the spatial dependencies might be irrelevant for the image under consideration, it is essential to select the relevant contextual information that have discriminative capabilities and block the irrelevant ones. A variant of RNNs called Long Short-Term Memory (LSTM) [124] cells have been used to spot salient keywords in sentences [116] and speech inputs [129] to learn context (i.e., topic) relevant information. The advanced gating mechanisms inside the LSTM cell can regulate the information flowing into and out of the cell [127]. The extracted salient contextual information can further enhance the discriminative power of the learned local feature representations by removing irrelevant, noisy and background information.

However, for human re-identification, in the embedded feature space, the feature vectors of similar pairs (i.e., from the same subject) must be ‘close’ to each other while the feature vectors from dissimilar pairs should be distant to each other. To this end, we propose a siamese architecture based on LSTM cells. Siamese networks consist of two identical sub-networks joined at the output which are used for comparing two input fields [15]. For learning the network parameters, inputs are therefore given in the form of pairs. The network is optimized by a contrastive loss function [143]. The fundamental idea of the contrastive loss function is to ‘attract’ similar inputs towards each other and ‘repel’ dissimilar inputs. As a result, LSTM network can selectively propagate the context that can bring together the positive pairs and push apart the negative pairs. Though

the triplet ranking loss function has also been the basis of several human re-identification approaches [139, 173, 174], in the proposed architecture, the Siamese loss is employed as it explicitly optimizes the distance between positive pairs where as conventional triplet loss formulations [173, 174] optimize the objective such that the distance between negative pairs is larger than the distance between positive pairs. However, a recent approach [139] addresses this issue for the task of human re-identification. Additionally, the triplet loss requires much more careful sampling of the data triplets compared to sampling pairs so as to eliminate trivial losses that does not contribute to the network learning. This means that the sampled pairs must satisfy the criteria that the distance between positive pairs is lesser than that of the negative pair for the same anchor image. However, while sampling pairs, such a constraint does not exist even though the distance between negative pairs must be lesser than the margin.

The image is divided into several horizontal stripes and is represented as a sequence of image regions following [13] and starting from the first horizontal stripe, the LSTM cell progressively takes each of the horizontal stripes as inputs and decides whether to retain the information from the current input or discard it based on the information it captured from the current and previous inputs. Similarly, the LSTM cell can hide (or release) the contents of the memory from (or to) the other components of the network at each step. Detailed experimental evaluation of our proposed approach was conducted and the proposed approach outperforms a baseline without LSTM units and achieve promising results compared to the state-of-the-art algorithms. We also provide intuitive visualizations and explanations to demonstrate the internal interactions of LSTM cells and prove the effectiveness of our approach. The major contributions of this chapter can be summarized as follows.

- In this work, the LSTM is adapted to human re-identification task that can leverage the contextual information to enhance the discriminative capability of the local features. This significantly differs from the traditional methods that perform feature extraction locally and independent of other regions. Although adjacency constrained local patch based search approaches [4, 31, 137] as well as local patch clustering based approaches [137, 145] have been proposed before, the main objective in [4, 31, 145] is to obtain a saliency map for each pedestrian image which can

be discriminative. The mid-level filter learning approach proposed in [137] aims to find effective patch clusters to improve the discriminative and generalization capability of the image representation. However, the proposed model aims to explicitly incorporate the spatial information (i.e. information from other parts of the image) using a deep feature learning based approach to improve the discriminative capability of the local features.

- A novel siamese LSTM architecture optimized by the contrastive loss function is proposed for learning an embedded feature space where similar pairs are closer to each other and dissimilar pairs are distant from each other.
- The proposed approach achieves better performance when compared to a baseline algorithm (without LSTM units) as well as promising results when compared to several state-of-the-art algorithms for human re-identification. We also evaluate the multiplicative internal interactions of the LSTM cells and provide intuitive visualizations to demonstrate the effectiveness of our approach.

The rest of this chapter is organized as follows. First a brief overview of the applications of Recurrent Neural Networks in computer vision is given in Section 2.2.1.2. The proposed framework is introduced in Section 4.2. Section 4.3 provides the detailed experimental settings and results and in Section 4.4, the internal mechanisms of the LSTM unit is visualized and analyzed. Section 4.5 concludes this chapter.

4.2 Proposed Framework

Overview: The goal of our model is to match images of same pedestrians obtained from different surveillance cameras. The proposed siamese architecture consists of two copies of the Long-Short Term Memory network sharing the same set of parameters. The network is optimized based on the contrastive loss function. Figure 4.2 illustrates the proposed siamese LSTM architecture. Below, we explain our motivation through the introduction of the Long-Short Term Memory networks. Our proposed siamese architecture is further explained in detail with the optimization methodologies.

Figure 4.2: Proposed siamese LSTM architecture.

4.2.1 Learning Contextual Dependency using LSTM

RNN architectures have been previously used in [12, 13, 120] to model the spatial dependency and extract more discriminative features for image classification and scene labeling. While encoding such spatial correlations, traditional RNNs do not have the flexibility to selectively choose relevant contexts. Our work is motivated by the previous works [116, 129] which have proven that the LSTM architectures can spot salient keywords from sentences and speech inputs. The internal gating mechanisms in the LSTM cells can regulate the propagation of certain relevant contexts, which enhance the discriminative capability of local features. With these key insights, we propose a siamese LSTM network with pairs of images as inputs for the human re-identification task. The network is modeled in such a way that it accepts the inputs (horizontal stripes) one-by-one and progressively capture and aggregate the relevant contextual information. The contrastive loss function is used to optimize the network parameters so that the learned discriminative features can successfully bring the positive pairs together and push apart the negative pairs. Below, we explain the dynamics of a single layer LSTM architecture without peephole connections.

4.2.1.1 Long Short-Term Memory Networks

LSTM networks were introduced to address the vanishing gradient problem associated with RNNs and a more complete version was proposed in [144, 175, 176] with forget gate, peephole connection and full BPTT training. For a detailed formulation and explanation of LSTM architecture and mathematical equations, the reader is referred to Chapter 2,

Section 2.2.1.2. As discussed in Chapter 2, Section 2.2.1.2, the internal gating mechanisms inside the LSTM cell can regulate the information flow through the network. The *input* gate, the *output* gate and the *forget* gate can selectively propagate the information flow into and out of the LSTM unit, or reset its state while learning the contextual dependencies.

4.2.2 The Proposed Siamese LSTM Architecture

For human re-identification, the objective is to retrieve a set of matching gallery images for a given query image. Therefore, we develop a siamese network to take pairs of images as inputs and learn an embedding, where representations of similar image pairs are closer to each other while dissimilar image pairs are distant from each other. All the images in the dataset are paired based on the prior knowledge of the relationships between the images (i.e., similar or dissimilar pairs). Consider an image pair (I^p, I^q) as shown in Figure 4.2, corresponding to the i^{th} pair of images ($i = \{1, 2, \dots, N_{pairs}\}$). N_{pairs} indicates the total number of image pairs. Let $Y^i \in [0, 1]$ be the label of the i^{th} pair. $Y^i = 0$ indicates that the images are similar and $Y^i = 1$ indicates that they are dissimilar. Input features are first extracted from the images. Following previous works [1, 22, 34], the input image is divided into several horizontal stripes and thus, treated as a spatial sequence as opposed to a temporal sequence typically observed in acoustic or natural language inputs. Dividing the image into horizontal rows has the advantage of translational invariance across different view points. We use r as a suffix to denote the local features at a particular region (eg: $\mathbf{x}_r$; $r = \{1, 2, \dots, R\}$). R indicates the total number of regions (rows).

Dividing the image into rows has the advantage of translational invariance, which is important for human re-identification and is a commonly adopted strategy to represent image features [1, 22].

4.2.2.1 Input Features

We extract the state-of-the-art features (LOMO) and Color Names [71] features from the images regions, corresponding to rows.

- **Local Maximal Occurrence (LOMO):** To extract the LOMO features, first, Color histogram feature and SILTP features [38] are extracted over 10×10 blocks with an overlap of 5 pixels. The feature representation of a row is obtained by maximizing the local occurrence of each pattern at the same horizontal location. For an image with 128×64 pixels, this yields 24 rows. Following the same settings, the features are extracted at 3 different scales which resulted in 24, 11 and 5 rows per image.
- **Color Names (CN) :** Features are extracted for 4×4 blocks with a step-size of 4. Further a row-wise feature representation is obtained by combining the BoW features along the same horizontal location. We follow the same settings and code-book provided in [1] for feature extraction. The final feature representation yields 16 rows for each image with the size of 128×64 .

4.2.2.2 Objective Function

Let the input features from the r^{th} region (row) from the image pairs (I^p, I^q) be $\mathbf{x}_r^p$ and $\mathbf{x}_r^q$ respectively. As shown in Figure 4.2, the input features $\mathbf{x}_r^p$ and $\mathbf{x}_r^q$ are fed into parallel LSTM networks. Note that dividing the image into 6 parts (i.e., rows) is merely for illustration and is not exactly the same in our experimental settings. Each of these LSTM networks process the input sequentially and the hidden representations $\mathbf{h}_r^p$ and $\mathbf{h}_r^q$ at each step are obtained using Eq. (2.5) presented in Chapter 2. The hidden representation at a particular step is a function of the input at the current step and the hidden representation at the previous step. For example, $\mathbf{h}_r^p$ is a function of $\mathbf{x}_r^p$ and $\mathbf{h}_{r-1}^p$. In the proposed architecture, we use a single layer LSTM. Therefore, the hidden representations $\mathbf{h}_r^p$ and $\mathbf{h}_r^q$ obtained from the LSTM networks are used as the input representations for the rest of the network.

Once the hidden representations from all the regions are obtained, they are combined to obtain $\mathbf{s}_p$ and $\mathbf{s}_q$ as shown below.

$$\mathbf{s}_p = \mathbf{W}_M^T [(\mathbf{h}_1^p)^T, \dots, (\mathbf{h}_r^p)^T, \dots, (\mathbf{h}_R^p)^T]^T; \quad r = 1, 2, \dots, R \quad (4.1)$$

$$\mathbf{s}_q = \mathbf{W}_M^T [(\mathbf{h}_1^q)^T, \dots, (\mathbf{h}_r^q)^T, \dots, (\mathbf{h}_R^q)^T]^T; \quad r = 1, 2, \dots, R \quad (4.2)$$

where $\mathbf{W}_M \in \mathbb{R}^{(R*n) \times (R*n)}$ is the transformation matrix. $[.]^T$ indicates the transpose operator. The objective of the framework is that $\mathbf{s}_p$ and $\mathbf{s}_q$ should be closer to each other

if they are similar and far from each other if they are dissimilar. The distance between the samples, $\mathbf{s}_p$ and $\mathbf{s}_q$ ($D_s(\mathbf{s}_p, \mathbf{s}_q)$) can be given as follows:

$$D_s(\mathbf{s}_p, \mathbf{s}_q) = \|\mathbf{s}_p - \mathbf{s}_q\|_2 \quad (4.3)$$

Once the distance between the representations $\mathbf{s}_p$ and $\mathbf{s}_q$ is obtained, it is given as the inputs to the contrastive loss objective function. It can be formally written as:

$$L(\mathbf{s}_p, \mathbf{s}_q, Y^i) = (1 - Y^i) \frac{1}{2} (D_s)^2 + (Y^i) \frac{1}{2} \{ \max(0, m - D_s) \}^2 \quad (4.4)$$

where $m > 0$ denotes a margin which acts as a boundary (with the radius m). The intuition behind this loss function is that dissimilar pairs must be separated by a distance defined by m and similar pairs must be as close as possible (i.e., distance tends to 0). For more details regarding the loss function, we refer the interested readers to [143]. The total loss can be obtained by taking the sum of the losses for all pairs .

4.2.2.3 Network Training

The overall loss is minimized so that similar pairs are closer to each other and dissimilar pairs are separated by m . For positive pairs, the label is set as ‘0’ and for negative pairs, the label is set as ‘1’ following previous works [143]. The system is trained by back-propagating the gradients of Eq. (4.4) with respect to $\mathbf{s}_p$ and $\mathbf{s}_q$ through the network. While generating the input image pairs, we do not consider all the negative images for a particular identity as it results in a biased dataset. Following the previous works [138], the number of hard-negatives sampled is twice the number of positive pairs per image. To sample the hard-negatives, we consider the closest matching images in the input feature space (not in the raw image space). Even-though learning frameworks generalize better when trained with large datasets, we perform the training without any data augmentation or fine-tuning operations.

4.2.2.4 Optimization

We use the mini-batch stochastic gradient descent method with the batch size of 100 pairs. Weight parameters ($\mathbf{W}_L$ and $\mathbf{W}_M$) are initialized uniformly in the range of $[-a, a]$ where $a = \sqrt{1/(\text{input size}(d) + \text{hidden size}(n))}$ [177]. In traditional RNN/LSTM architectures, gradients are computed using the BPTT algorithm [131]. In the proposed siamese LSTM architecture, the gradients with respect to the feature vectors extracted from each pair of images are calculated and then back propagated for the respective branches independently using BPTT. As the parameters in each branch are shared, the gradients of the parameters are summed up and then the weights are updated. RMSProp [178] per parameter adaptive update strategy is used to update the weight parameters. Following the previous works [113, 126], we keep the decay parameter as 0.95 and clip the gradients element-wise at 5. These settings are fixed and found to be robust for all the datasets. The only tuned parameters were the hidden vector size (n), learning rate (lr) and margin (m) for the contrastive loss function (see Eq. (4.4)). Training is performed for a maximum of 20 epochs with an early stopping scheme if the cross-validation performance is found to be saturating. The training/validation curves of the proposed baseline algorithm as well as the Siamese LSTM algorithm on the CUHK03 dataset is shown in Figure 4.4. It can be seen that the proposed approach achieves lower validation error compared to the baseline algorithm indicating that the learned model has better generalization capabilities.

4.2.2.5 Testing

During the testing process, the local features for all the query and gallery images are extracted and mapped using the proposed framework to obtain the corresponding representations $\mathbf{s}_p$ ($p = \{1, \dots, N_{query}\}$) and $\mathbf{s}_q$ ($q = \{1, \dots, N_{gallery}\}$), where N_{query} and $N_{gallery}$ denote the total number of images in the query set and the gallery set, respectively. The total number of query-gallery pairs will be $N_{query} \times N_{gallery}$. The final decision is made by comparing the Euclidean distance (i.e., matching scores) between all $\mathbf{s}_p$ and $\mathbf{s}_q$, $D_s(\mathbf{s}_p, \mathbf{s}_q)$. When using multiple features, the matching scores obtained per query image with respect to all the gallery images for each feature are rescaled in the range of 0 – 1 and then averaged. The final best match is the gallery image that has the least Euclidean distance based on the averaged scores.

4.3 Experiments

In this section, we present a comprehensive evaluation of the proposed algorithm by comparing it with a baseline algorithm as well as several state-of-the-art algorithms for human re-identification. In most existing human re-identification works, the Cumulative Matching Characteristics (CMC) results were reported. However, in [1], human re-identification is treated mainly as a retrieval problem, so the rank 1 accuracy (R1 Acc) and the mean average precision (mAP) are used for performance evaluation. For a fair comparison with the baseline as well as the state-of-the-art algorithms, we report both CMC and mAP on all three datasets.

Baseline: To evaluate the performance contribution of the proposed LSTM based siamese network, we implement a baseline method without using LSTM, i.e., with a mapping $\mathbf{W}$ alone. Features from all rows were concatenated and given as input in contrast to concatenating the hidden features from LSTM. Formally, the equations for obtaining $\mathbf{s}_p$ and $\mathbf{s}_q$ using a single layer baseline can be given as follows:

$$\mathbf{s}_p = f(\mathbf{W}^T[(\mathbf{x}_1^p)^T, \dots, (\mathbf{x}_r^p)^T, \dots, (\mathbf{x}_R^p)^T]^T) \quad (4.5)$$

$$\mathbf{s}_q = f(\mathbf{W}^T[(\mathbf{x}_1^q)^T, \dots, (\mathbf{x}_r^q)^T, \dots, (\mathbf{x}_R^q)^T]^T) \quad (4.6)$$

where $f(\cdot)$ is a non-linear sigmoid activation function and $\mathbf{W}$ is the parameter matrix that is to be learned. The above system was optimized based on the same contrastive loss function in Eq. (4.4). We also report the results using a multi-layer baseline which can be obtained by extending the above framework to multiple layers.

4.3.1 Datasets and Experimental Settings

The experiments were conducted on 3 challenging human re-identification datasets, Market-1501 [1], CUHK03 [19] and VIPeR [16] datasets introduced in Section 1.4. Following the standard evaluation protocols in [1], the Market-1501 dataset is split into 751 identities for training and 750 identities for testing. We provide the multi-query evaluation results. For multi-query evaluation, the matching scores for each of the query images from one subject are averaged. For CUHK03 dataset, twenty random splits are provided in [19] and the average results over all splits are reported. For multi-query evaluation,

the matching scores from each of the query images belonging to the same identity are averaged. Lastly, for the VIPeR dataset, the data is split randomly into equal halves and cross camera search is performed to evaluate the algorithms.

Table 4.1 shows the performance comparison of the proposed algorithm with the baseline algorithm. It can be seen that the proposed LSTM architecture outperforms the single-layer and multi-layer baseline algorithms for all the datasets. Results indicate that feature selection based on the contextual dependency is effective for re-identification tasks. The Rank 1 performance on VIPeR dataset for the 3-layer baseline is lower compared to the 2 layer and 1 layer approach. We believe that this may be due to over-fitting as the dataset is smaller compared to the CUHK03 and Market-1501 datasets. Comparison with the state-of-the-art algorithms is shown in Table 4.4, 4.5 and 4.6. For a fair evaluation, we compare our results to only individual algorithms and **not** to ensemble methods [137, 179]. Some qualitative results are shown in Figure 4.5. It can be seen that the proposed algorithm retrieves visually similar images thereby improving the re-identification rate and mean average precision.

4.3.2 Parameter Tuning

All the parameters are tuned by conducting cross-validation on the training data. Figure 4.3 shows the cross validation results using the LOMO [22] features on the Market-1501 dataset. The Rank 1 Accuracy (R1 Acc) as well as the mean average precision (mAP) is indicated in the graphs showing the cross-validation curves. From Figure 4.3 (a), it can be observed that when setting the LSTM hidden vector dimension larger than 25, there is no significant improvement in the validation performance. Similarly, from the Figure 4.3 (b), it can be observed that the validation performance drops beyond margin $m = 0.75$. For our experiments, we set $m = 0.5$ as there is a slight advantage in the validation performance. For tuning the learning rate, we conduct a log-uniform sampling in the range $[10^{-9}, 10^{-1}]$. For more detailed information on hyper-parameter search methods, we refer the interested readers to [127].

Table 4.1: The CMC and mAP on the Market-1501, CUHK03 and VIPeR datasets.

Dataset	Market 1501		CUHK03				VIPeR			
	Rank 1	mAP	Rank 1	Rank 5	Rank 10	mAP	Rank 1	Rank 5	Rank 10	mAP
Baseline (LOMO) - 1 Layer	46.9	21.3	49.1	76.0	85.3	40.1	35.8	62.3	75.0	42.8
Baseline (LOMO) - 2 Layer	47.8	23.9	50.4	77.6	85.9	40.9	36.3	63.6	75.3	42.9
Baseline (LOMO) - 3 Layer	48.4	24.8	51.1	78.3	86.1	41.8	34.8	63.3	75.1	41.5
LSTM (LOMO) - 1 Layer	51.8	26.3	55.8	79.7	88.2	44.2	40.5	64.9	76.3	45.9
Baseline (LOMO + CN) - 1 Layer	52.1	27.1	51.6	76.6	85.8	42.1	36.1	64.9	75.6	43.0
LSTM (LOMO + CN) - 1 Layer	61.6	35.3	57.3	80.1	88.3	46.3	42.4	68.7	79.4	47.9

4.4 Analysis

4.4.1 Internal Mechanisms of the LSTM cell

Figure 4.5 (a) shows two example queries from a testset of the VIPeR dataset and the input gate activations of the **query** image. The retrieved matches for the query image are shown in Figure 4.5 (b). To clearly distinguish between the different values, the gate activations are given as **heat maps**. From the ‘response’ of the gates to certain inputs, we would like to answer the question whether the LSTM can select and ‘memorize’ the relevant contextual information and discard the irrelevant ones. The optimal hidden dimensions (n) for the VIPeR dataset was found to be 50 for the LOMO features. Therefore, the gate activations are 50 dimensional vectors whose values range from 0 – 1. In Figure 4.5 (a), we show the L2 norm values of the gate activations at each step (24 steps in total corresponding to 24 local regions of the image). The L2 norm values are represented as a heat map where values close to 1 (right saturated - information is propagated) is represented by darker shades of red and values closer to 0 (left saturated - information is blocked) by deeper shades of blue. Kindly note that the $L2$ norm value is merely for better illustration and is not actually used in the system.

4.4.1.1 Example 1

We begin with the first example shown in Figure 4.5. Figure 4.6 (a) shows the first query given in Figure 4.5, the internal gate activations (with its $L2$ norm values) and the top 3 retrieved matches are shown in Figure 4.6 (b). Figure 4.6 (c), (d) and (e) shows the actual input, output and forget gate values and the $L2$ Norm value computed

and displayed for better illustration. The mechanisms shown are for the LOMO features extracted at the original scale (i.e., original size of the image). Therefore, as mentioned in Section 4.2.2, this results in 24 rows. The optimal hidden dimensions of the LOMO features was found to be 50 for VIPeR dataset. Therefore, the original input, forget and output gate activations shown in Figure 4.6 has 24 rows along the y-axis and 50 dimensions along the x-axis.

Based on Figure 4.7, we give the step by step explanation by taking sections of rows from the gate activations and the corresponding regions from the image.

Rows 1-3 : Rows 1 – 3 constitutes mostly the head region and some background regions as shown in Figure 4.7. These regions may be unimportant and less discriminative for re-identification in most cases. It can be seen that the values of the input, forget and the output gates are relatively low. Thus the input gates ‘attenuate’ the propagation of information flow from the current input. Since the forget gate values are low, the contents of the memory cell are erased indicating that the network does not ‘memorize’ information from the previous inputs. Similarly, output gate being low indicates that the cell contents are ‘blocked’ from being revealed at the output.

Rows 4 – 10 : Rows 4 – 10 constitutes mostly the upper part of the body which can contain the discriminative information for re-identification in several cases. It can be seen that the values of the input, forget and the output gates are relatively high compared to the previous rows. Thus the input gates ‘allow’ the propagation of information flow from the current input. Since the forget gate values are high, the contents of the memory cell are updated as it goes from step 4 – 10 indicating that the network can ‘memorize’ the contextual information from the previous inputs. Similarly, output gate being high indicates that the cell contents are ‘propagated’ through the rest of the network. Similar observations can be seen for Rows 13 – 14 as well even though it is not as strong as the Rows 4 – 10.

4.4.1.2 Example 2

The second example is shown in Figure 4.8 below (corresponds to the second query in Figure 4.5). Figure 4.8 (a) shows the query image and the $L2$ Norm of the Input Gate, Forget Gate and the Output Gate and Figure 4.8 (b) shows the retrieved matches at

Rank 1, Rank 2 and Rank 3. In Figure 4.8 (c), (d) and (e), the actual input, output and forget gate values and the $L2$ Norm value computed and displayed for better illustration. Similar to the above example, it can be observed that the input, forget and the output gate values are relatively low for rows 1 – 6. However, for most of the regions corresponding to the body of the subject, the activations are relatively high indicating that the network memorizes and propagates the relevant contextual information. From these observations, it can be inferred that the proposed siamese LSTM architecture can enhance the discriminative capability of the feature representations by selectively propagating the relevant contextual information. To explain this more clearly, the input gate values of the query, the Rank 1 retrieved match and the Rank 2 retrieved match is shown in Figure 4.9. It can be seen that for the Rank 1 image, input gate values selectively propagate the information mostly from the upper part of the body as well as the middle part of the body and this follows somewhat similar to that of the input query. However, for the rank 2 retrieved image, some of the information from the lower part of the body along with the upper and middle part are also selected by the input gate. Comparing these selected information, it can be seen that the LSTM gating unit learns to selectively propagate the discriminative information that can bring together the positive pairs and push apart the negative pairs.

4.4.1.3 More Successful and Failure cases

More examples of successful and failed cases are shown in Figures 4.10 and 4.11. We omit the $L2$ norm values and show the actual values of the input, forget and output gates for reference. The correct match is shown in green bounding boxes. Similar to the examples 1 and 2 in Figure 4.6 and 4.8, we can see that, for the successful cases (Figure 4.10) the input, forget and the output gates selectively propagate (where the heat map appears more reddish) certain discriminative information from the query images and the retrieval happens based on the propagated contextual information. However, it was observed that the network fails when the appearance of certain query and gallery images are too close to each other. For example, in Figure 4.11, even though the network propagates certain information from the query images, the retrieved gallery images follow relatively very similar visual appearance thus making it difficult for the network to accurately

retrieve the correct matches. To distinguish such hard-negatives from the positive pairs, the network has to focus on subtle patterns which can clearly differentiate the positive match from the other hard-negative matches. For example, in Figure 4.11, the fine texture pattern in the upper part of the body for the query image in row 3, the cap for the query image in row 2 etc, can clearly distinguish the actual match from the Rank 1 retrieved match. The gating mechanisms also fail to extract and propagate such information from the query images as can be seen from the heat maps for the input, forget and the output gates. However, for the successful matches in Figure 4.10, it can be seen that though there exist similarities in the visual appearance of the retrieved matches, the discriminative information are clearer than those of the failure cases.

4.4.2 Data Augmentation

Data augmentation is an extensively used technique especially in deep networks to improve the generalization capability of the learned network. To study if data augmentation can be beneficial for the proposed learning framework, experiments were conducted on the CUHK03 dataset by artificially augmenting the training set images. Specifically, data augmentation is performed by flipping the images, and doing random crops. The results on the CUHK03 dataset using LOMO features extracted at the original scale are reported in Table 4.2 and it can be seen that by using data augmentation, the results of the proposed Siamese LSTM architecture can be improved.

Table 4.2: Performance Comparison of proposed LSTM architecture with and without data augmentation on the CUHK03 dataset.

Method	Rank 1	Rank 5	Rank 10
Ours	52.4	78.9	87.1
Ours with augmentation	54.6	81.8	89.4

4.4.3 Fusion with XQDA

The proposed Siamese LSTM architecture aims to extract the contextual dependency between different parts of the image and selectively propagate certain relevant information from the inputs. Extracting the contextual dependency cues can be considered complementary to other approaches as majority of them focus on extracting richer local information or advanced metric learning formulation. To study this aspect, the proposed Siamese LSTM architecture was fused with the LOMO-XQDA [22] algorithm which is one of the state-of-the-art works in human re-identification. For this experiment, the LOMO features extracted at the original scale were used for both the Siamese LSTM as well as the LOMO-XQDA algorithm. Experiments were conducted on the CUHK03 dataset and the results are reported in table 4.3. It can be seen that the results of proposed algorithm can be improved by combining with LOMO-XQDA algorithm.

Table 4.3: Performance Comparison of proposed LSTM framework and the fusion of the proposed LSTM framework with LOMO-XQDA algorithm on the CUHK03 dataset.

Method	Rank 1	Rank 5	Rank 10
Ours	52.4	78.9	87.1
Ours + LOMO-XQDA [22]	60.3	84.5	92.5

Table 4.4: Performance Comparison of state-of-the-art algorithms for the Market-1501 dataset. Results for [3] and [4] are taken from [1].

Method	Rank 1	mAP
SDALF [3] - CVPR 2010	20.53	8.20
eSDC [4] - CVPR 2013	33.54	13.54
BoW + HS[1] - ICCV 2015	47.25	21.88
DNS [166] - CVPR 2016	71.56	46.03
Ours	61.60	35.31

4.4.4 State-of-the-art Comparison

Table 4.4, 4.5 and 4.6 shows the state-of-the-art comparisons on the Market-1501, CUHK03 and VIPeR datasets respectively. For Market-1501 dataset, a recent metric learning approach [166] outperforms ours and for CUHK03 dataset, several recent approaches such as [141, 167] outperforms ours. However, we believe that the above approaches can be complementary to our approach as the main contribution in this paper is on improving the discriminative capability of the local-features using contextual information. In [141], an advanced ConvNet architecture, the GoogleNet is proposed for human re-identification where as in [167], a regression framework is proposed for fast and scalable person re-identification. For VIPeR dataset, several recent approaches [139, 164–168] outperform our approach. We believe that the reason is lack of positive pairs per image (only 1) and also the lack of total number of distinct training identities compared to other larger datasets. However, to improve the performance on feature learning approaches such as ours, transfer learning from larger datasets or data-augmentation can be employed.

Table 4.5: Performance Comparison of state-of-the-art algorithms for the CUHK03 dataset.

Method	Rank 1	Rank 5	Rank 10
SDALF [3] - CVPR 2010	4.9	21.0	31.7
ITML [27] - ICML 2007	5.14	17.7	28.3
LMNN [28] - JMLR 2009	6.25	18.7	29.0
eSDC [4] - CVPR 2013	7.68	22.0	33.3
LDML [180] - ICCV 2009	10.9	32.3	46.7
KISSME [29] - CVPR 2012	11.7	33.3	48.0
FPNN [19] - CVPR 2014	19.9	49.3	64.7
BoW [1] - ICCV 2015	23.0	45.0	55.7
BoW + HS [1] - ICCV 2015	24.3	-	-
ConvNet [138] - CVPR 2015	45.0	75.3	55.0
LX [22] - CVPR 2015	46.3	78.9	88.6
MLAPG [30] - ICCV 2015	51.2	83.6	92.1
SS-SVM [164] - CVPR 2016	51.2	80.8	89.6
SI-CI [140] - CVPR 2016	52.2	84.3	92.3
DNS [166] - CVPR 2016	54.7	84.8	94.8
DGDropout [141] - CVPR 2016	75.3	-	-
HER [167] - BMVC 2016	65.2	92.2	96.8
Ours	57.3	80.1	88.3

4.5 Conclusion and Future Works

In this chapter, a novel siamese LSTM architecture was introduced for human re-identification. The network can learn the spatial dependencies between local regions of the image and selectively propagate relevant contextual information to enhance the discriminative capacity of the local features. To achieve the aforementioned task, the proposed approach exploits the powerful multiplicative interactions within the LSTM cells. By examining the activation statistics of the input, forget and output gating mechanisms in the LSTM

Table 4.6: Performance Comparison of state-of-the-art algorithms using an individual method for the VIPeR dataset.

Method	Rank 1	Rank 5	Rank 10
LFDA [76] - CVPR 2013	24.1	51.2	67.1
eSDC [4] - CVPR 2013	26.9	47.5	62.3
Mid-level [137] - CVPR 2014	29.1	52.3	65.9
SVMML [5] - CVPR 2013	29.4	63.3	76.3
VWCM [26] - ECCVW 2014	30.7	63.0	76.0
SalMatch [31] - ICCV 2013	30.2	52.3	65.5
QAF [160] - CVPR 2015	30.2	51.6	62.4
SCNN [142] - ICPR 2014	28.2	59.3	73.5
ConvNet [138] - CVPR 2015	34.8	63.7	75.8
CMWCE [23] - ICPR 2014	37.6	68.1	81.3
SCNCD [24] - ECCV 2014	37.8	68.5	81.2
LX [22] - CVPR 2015	40.0	68.1	80.5
PRCSL [161] - ICCV 2015	34.8	68.7	82.3
MLAPG [30] - ICCV 2015	40.7	69.9	82.3
MT-LORAE [162] - ICCV 2015	42.3	72.2	81.6
Semantic Representation [163] - CVPR 2015	41.6	71.9	86.2
DGDropout [141] - CVPR 2016	38.6	-	-
SI-CI [140] - CVPR 2016	35.8	67.4	83.5
SS-SVM [164] - CVPR 2016	42.7	-	84.3
MCP-CNN [139] - CVPR 2016	47.8	74.7	84.8
HGD [165] - CVPR 2016	49.7	79.7	88.7
DNS [166] - CVPR 2016	51.7	82.1	90.5
HER [167] - BMVC 2016	53.0	79.8	89.6
SCSP [168] - CVPR 2016	53.5	82.6	91.5
Ours	42.4	68.7	79.4

cell, we show that the network can selectively allow and block the context propagation and enable the network to ‘memorize’ important information. Our approach is evaluated on several challenging real-world human re-identification datasets and it consistently out-

performs the baseline and achieves promising results compared to the state-of-the-art. Partial work presented in this chapter has been published in [181].

LSTM architectures can successfully model the spatial context and feature selection based on the contextual information. However, the proposed framework lack the capability to model the pairwise relations. In conventional siamese architectures, the representations are compared only once they are matured, i.e. at the final layer. However, this setting may result in capturing fine patterns from the lower layers of the network that may be essential to enhance the discriminative capability of the network. In Chapter 5, a new Convolutional Neural Network model is introduced to model the pairwise relation and enhance the discriminative capability by feature selection. A Matching Gate is proposed to selectively propagate shared information from image pairs from the middle layers to the final layers of the network. The algorithm is also modeled in an end-to-end fashion eliminating the need of hand-crafted features.

Validation Curves

(a)

(b)

Figure 4.3: Cross-validation curves: (a) hidden size (n) vs Validation performance on Market-1501 dataset. (b) the margin (m) vs validation performance on the Market-1501 dataset.

Figure 4.4: Training/Validation curves on the CUHK03 dataset.

Figure 4.5: Qualitative Results showing the internal gating mechanism of the Siamese LSTM architecture

Figure 4.6: Example 1 of a test image from VIPeR dataset

Figure 4.7: The different rows of the query image and the corresponding input gate, forget gate and the output gates for example 1.

Figure 4.8: Example 2 of a test image from VIPeR dataset.

Figure 4.9: The input gate values of the query, the Rank 1 retrieved match as well as the Rank 2 retrieved match. See text for details.

Successful Matches

Figure 4.10: More examples of successful matches

Some failed cases

Figure 4.11: Some failed cases

Chapter 5

Gated Siamese Convolutional Neural Network

In the previous chapters, the benefits of invariant features as well as learning the contextual dependencies for extracting discriminative local features have been demonstrated. However, in these frameworks, the feature representations of the images are compared only when they mature, i.e. only the feature representations from the last layer is used for comparing two images. In this setting, the network is at risk of failing to extract and propagate subtle local patterns that may be essential to discriminate positive pairs from hard-negative pairs. In this chapter, an end-to-end trainable Siamese Convolutional Neural Network (S-CNN) is initially proposed for the human re-identification task. Further, to address the aforementioned problem with the S-CNN architecture, a differentiable gating mechanism is proposed to extract fine local patterns from the middle layers of the network and propagate to the top layers of the network by amplifying them.

5.1 Introduction

Convolutional Neural Networks (CNNs) have shown remarkable success in several computer vision tasks such as image/scene classification [14, 79–82, 111], object recognition [79–81], semantic segmentation [12, 112], image/video captioning [113–115] and a lot more. With the resurgence of Convolutional Neural Networks (CNNs) in the past decade [111, 182, 183], several deep learning methods [19, 138, 142] were proposed for human re-identification. Most of the frameworks are designed in a siamese fashion that integrates the tasks of feature extraction and metric learning into a single framework. The central idea behind majority of the Siamese Convolutional Neural Network (S-CNN) is to learn an embedding where similar pairs (i.e. images belonging to the same identity) are close to each other and dissimilar pairs (i.e. images belonging to different identities) are separated by a distance defined by a parameter called ‘margin’ [15, 143]. In this chapter, we first propose a baseline S-CNN architecture that can outperform majority of the deep learning architectures as well as other handcrafted approaches for human re-identification.

Though S-CNNs have shown incredible performance gains in human re-identification tasks, the major drawback of the S-CNN architecture is that it extract fixed representations for each image without the knowledge of the paired image. This setting results in a risk of failing to capture and propagate the local patterns that are necessary to increase the confidence level (i.e., reducing the distances) in identifying the correct matches. Figure 5.1 (a) and (b) shows two queries and the retrieved matches at the top 3 ranks using a S-CNN architecture. Even though there are obvious dissimilarities among the top 3 matches for a human observer in both the cases, the network fails to identify the correct match at Rank 1. For example, the patches corresponding to the ‘bag’ (indicated by red boxes) in Figure 5.1 (a) and the patches corresponding to the ‘hat’ (indicated by blue boxes) in Figure 5.1 (b) could be helpful to distinguish between the top retrieved match and the actual positive pairs. However, a network that fails to capture and propagate such finer details may not perform well in efficiently distinguishing positives from hard-negatives.

CNNs extract low-level features at the bottom layers and learn more abstract concepts such as the parts or more complicated texture patterns at the mid-level. Since the mid-level features are more informative compared to the higher-level features, the

Figure 5.1: Results obtained using a S-CNN. Red, Blue and Yellow boxes indicate some sample corresponding patches extracted from the images along the same horizontal row.

finer details that may be necessary to increase the similarity for positive pairs can be more evident at the middle layers. Hence, we propose a gating function to compare the extracted local patterns for an image pair starting from the mid-level and promote (i.e. to amplify) the local similarities along the higher layers so that the network propagates more relevant features to the higher layers of the network. Additionally, during training phase, the mechanisms inside the gating function also boost the back propagated gradients corresponding to the amplified local similarities. This encourages the lower and middle layers to learn filters to extract more locally similar patterns that discriminate positive pairs from negative pairs. Hereafter, we refer to the proposed gating function as ‘the Matching Gate’ (MG) and the S-CNN architecture with the MG as ‘Gated S-CNN’.

The primary challenge in developing the matching gate is that it should be able to compare the local features across two views effectively and select the common patterns. Due to pose change across two views, features appearing at one location may not necessarily appear in the same location for its paired image. Since all the images are resized to a fixed scale, it is reasonable to assume a horizontal row-wise correspondence. Therefore, the matching gate first summarizes the features along each horizontal stripe for a pair of images and compares it by taking the Euclidean distance along each dimension of the obtained feature map. Once the distances between each individual dimensions are obtained, a Gaussian activation function is used to output a similarity score ranging from 0 – 1 where 0 indicates that the stripe features are dissimilar and 1 indicating that the stripe features are similar. These values are used to gate the stripe features and finally, the gated features are added to the input features to boost them thus giving

more emphasis to the local similarities across view-points. Although previous works have proposed adjacency constrained local patch search [4, 31, 137] as well as patch clustering based models [137, 145], the main aim in [4, 31, 145] is to construct a saliency map while discriminative and generalizable mid-level features are learned in [137]. Though the matching gate can be treated as a saliency modeling approach, it is modeled as a component in a deep learning algorithm to explicitly boost the subtle local similarities. Matching gate can also be treated as a neuron selection algorithm which is closely related to the DGD model [141], however the proposed matching gate aims to boost subtle local similarities based on gating the neurons while the central idea in [141] is to selectively activate the neurons corresponding to different re-identification domains.

The proposed approach does not require any part-level correspondence annotation between image pairs during the training phase as it directly compares the extracted mid-level features along corresponding horizontal stripes. Additionally, the proposed matching gate is formulated as a differentiable parametric function to facilitate the end-to-end learning strategy of typical deep learning architectures. To summarize, the major contributions of the proposed work are:

- We propose a baseline siamese convolutional neural network architecture that can outperform majority of the existing deep learning frameworks for the human re-identification task.
- To incorporate run time feature selection and boosting into the S-CNN architecture, we propose a novel matching gate that can boost the common local features across two views. This encourages the network to learn filters that can extract subtle patterns to discriminate hard-negatives from positive pairs. The proposed matching gate is differentiable to facilitate end-to-end training of the S-CNN architecture.
- Detailed experimental analysis on human re-identification datasets prove the effectiveness of our approach. The proposed framework also achieves promising results compared to the state-of-the-art algorithms.

The rest of this chapter is organized as follows. In Section 5.2, we introduce the proposed S-CNN and Gated S-CNN models. Section 5.3 covers the detailed experimental

Table 5.1: Proposed Baseline Siamese Convolutional Neural Network architecture.

Input	Conv Block - P2	Max Pool	Conv Block - P1	Max Pool	Conv Block - P1	Max Pool	Conv Block	Conv Block	Conv Block	Conv Block
128×64	$5 \times 5 \times 3 \times 32$	2×2	$3 \times 3 \times 32 \times 50$	2×2	$3 \times 3 \times 50 \times 32$	2×2	$1 \times 4 \times 32 \times 32$	$1 \times 3 \times 32 \times 32$	$1 \times 3 \times 32 \times 32$	$16 \times 1 \times 32 \times 150$

ConvBlock - Convolution $\rightarrow$ Batch Normalization $\rightarrow$ Parametric Rectified Linear Unit

P2 and P1 - zero padding the input with 2 pixels and 1 pixel on all sides respectively before convolution

results of the proposed approaches on challenging benchmarks for human re-identification. In Section 5.4, we give a detailed analysis of the proposed approaches by comparing it with different variants of S-CNN architectures. Section 5.5 provides the concluding remarks for this chapter.

5.2 Proposed Model

In this section, we first describe our baseline S-CNN architecture and further introduce the Matching Gate to address the limitations of the baseline S-CNN architecture.

5.2.1 Model Architecture

5.2.1.1 Baseline Siamese CNN Architecture:

The fundamental CNN architecture is modeled in a siamese fashion optimized by the contrastive loss function proposed in [143]. Table 5.1 summarizes the proposed Siamese CNN architecture. All the inputs are resized to a resolution of 128×64 and the mean image computed on the training set is subtracted from all the images. The description of the proposed S-CNN layers is as follows. First, we limit the number of pooling layers to only 3 so that it results in less information loss as the features propagate through the network. Second, we also use asymmetric filtering in layers 4 – 6 to preserve the number of rows at the output of the third layer while reducing the number of ‘columns’ progressively to 1. This strategy is inspired by the technique introduced in [22] in which the features along a single row is pooled to make the final feature map to a shape (number of rows) $\times 1$. It also helps to reduce the number of parameters compared to symmetric filters. Further, this feature map is fed into a fully connected layer which is the last layer of our network. Finally, we also incorporate some of the established state-of-the-art techniques to the proposed S-CNN architecture. As suggested in VGG-Net [81], we use smaller convolutional filters to reduce the number of parameters to be

learned while making the framework deeper. We also employ Batch Normalization [94] for standardizing the distribution of the inputs to each layer which helps in accelerating the training procedure. Parametric rectified linear unit (PReLU) [93] was used as the non-linear activation function as it has shown better convergence properties and performance gains with little risk of over-fitting. More results and analysis about the design choices are given in Section 5.4. The proposed S-CNN architecture outperforms majority of the existing approaches for human re-identification. However, as discussed in Section 5.1, the S-CNN model is not capable of adaptively emphasizing the local features that may be helpful to distinguish the correct matches from hard-negative pairs during run time. Therefore, we propose a matching gate to address this drawback. Below we give the details of the proposed module.

5.2.1.2 Matching Gate:

The proposed matching gate (MG) receives input activations from the previous convolutional block, compares the local features along a horizontal stripe and outputs a gating mask indicating how much more emphasis should be paid to each of the local patterns. Figure 5.2 illustrates the proposed final architecture with the gating function. The various components of the proposed MG are given below.

1. **Feature summarization:** The feature summarization unit aggregates the local features along a horizontal stripe in an image. This is necessary due to the pose changes of the pedestrian images across different views. For instance, as shown in Figure 5.1, the local features (indicated by red, blue and yellow boxes) appearing in one view may not be exactly at the same position in the other view, but it is very likely to be along the same horizontal region.

Let $\mathbf{x}_{r1} \in \mathbb{R}^{1 \times c \times h}$ be the input stripe features from the r^{th} row of a feature map at the input of the MG from one view point and $\mathbf{x}_{r2} \in \mathbb{R}^{1 \times c \times h}$ be the corresponding input stripe features from the other view point. Here, c denotes the number of columns and h denotes the depth of the input feature map. Given $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$, we propose to use a convolution strategy followed by the parametric rectified linear unit activation (PReLU) to summarize the features along the row resulting in feature

Figure 5.2: The proposed architecture which is a modified version of our baseline S-CNN.

vectors $\mathbf{y}_{r1}$ and $\mathbf{y}_{r2}$ respectively with dimensions $\mathbb{R}^{1 \times 1 \times h}$. The input features $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$, are convolved with filters $\mathbf{w} \in \mathbb{R}^{1 \times c \times h \times h}$ without any padding. This will compute the combination of different extracted patterns along each of the feature maps of $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$.

Mathematically, it can be expressed as

$$\mathbf{y}_{r1} = f(\mathbf{w} * \mathbf{x}_{r1}); \quad \mathbf{y}_{r2} = f(\mathbf{w} * \mathbf{x}_{r2}) \quad (5.1)$$

where ‘ $*$ ’ denotes the convolution operation and $f(\cdot)$ denotes the PReLU activation function. The bias is omitted in equation (5.1) for brevity. The parameters $\mathbf{w}$ and bias of the summarization unit can be learned along with the other parameters of the matching gate through back-propagation.

2. **Feature Similarity computation:** Once the features along a horizontal stripe are summarized across the two views, the similarity between them is computed. The

similarity is computed by calculating the Euclidean distance along each dimension ‘ h ’ of the summarized features. Computing the distance between each dimension is important as the gating function must have the flexibility to smoothly turn ‘on’ or turn ‘off’ each of the extracted patterns in the feature map. Once the distance is computed, a Gaussian activation function is used to obtain the gate values. The value of the Gaussian activation function varies from 0 – 1 and acts as a smooth switch for the input features. It also helps the function to be differentiable which is essential for end-to-end training of the S-CNN framework. Mathematically the gating value for each of the dimensions along row ‘ r ’ can be obtained as given below;

$$\mathbf{g_r}^i = \exp\left(\frac{-(\mathbf{y_{r1}}^i - \mathbf{y_{r2}}^i)^2}{\mathbf{p_i}^2}\right) \quad (5.2)$$

where $\mathbf{g_r}^i, \mathbf{y_{r1}}^i$ and $\mathbf{y_{r2}}^i$ denotes the i^{th} ($i = \{1, 2, \dots, h\}$) dimension of the gate values ($\mathbf{g_r}$), $\mathbf{y_{r1}}$ and $\mathbf{y_{r2}}$ respectively for the r^{th} row. The parameter $\mathbf{p_i}$ decides the variance of the Gaussian function and the optimal value can be learned during the training phase. It is particularly important to set a higher initial value for $\mathbf{p_i}$ to ensure smooth flow of feature activations and gradients during forward and backward pass in the initial iterations of the training phase. Further, the network can decide the variance of the Gaussian function for each dimension by learning an optimal $\mathbf{p_i}$.

3. **Filtering and Boosting the features:** Once the gate values ($\mathbf{g_r}$) are computed, each dimension along a row of the input is gated with the corresponding dimension of $\mathbf{g_r}$. The computed gate values will be of dimensions $\mathbb{R}^{1 \times 1 \times h}$ and is repeated c times horizontally to obtain $\mathbf{G_r} \in \mathbb{R}^{1 \times c \times h}$ matrix and further an element wise product is computed with the input stripe features $\mathbf{x_{r1}}$ and $\mathbf{x_{r2}}$. This will ‘select’ the common patterns along a row from the images appearing in both views. To boost these selected common patterns, the input is again added to these gated values. Mathematically, each dimension of the boosted output can be written as

$$\mathbf{a_{r1}}^i = \mathbf{x_{r1}}^i + \mathbf{x_{r1}}^i \odot \mathbf{G_r}^i \quad (5.3)$$

$$\mathbf{a_{r2}}^i = \mathbf{x_{r2}}^i + \mathbf{x_{r2}}^i \odot \mathbf{G_r}^i \quad (5.4)$$

$$\mathbf{G}_r^i = [\mathbf{g}_r^i, \mathbf{g}_r^i, \dots, \mathbf{g}_r^i]_{\text{repeated } c \text{ times}} \quad (5.5)$$

where $\mathbf{a}_{r1}^i, \mathbf{a}_{r2}^i, \mathbf{x}_{r1}^i, \mathbf{x}_{r2}^i, \mathbf{G}_r^i \in \mathbb{R}^{1 \times c \times 1}$. Once the boosted output $\mathbf{a}_{r1}$ and $\mathbf{a}_{r2}$ are obtained, we perform an $L2$ normalization across channels and the obtained features are propagated to the rest of the network. From Equations (5.3) and (5.4), we can understand that the gradients with respect to the ‘selected’ $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$ will also be boosted during the backward pass. To illustrate this, a network without the MG is considered. The $\mathbf{a}_{r1}^i$ and $\mathbf{a}_{r2}^i$ will remain the same as $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$ respectively as there is no explicit gating mechanism.

$$\mathbf{a}_{r1}^i = \mathbf{x}_{r1}^i \quad (5.6)$$

$$\mathbf{a}_{r2}^i = \mathbf{x}_{r2}^i \quad (5.7)$$

Therefore, the gradients of the loss function with respect to $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$ will also be the same as the gradients back-propagated from the loss layer for $\mathbf{a}_{r1}^i$ and $\mathbf{a}_{r2}^i$ respectively. However, for the network with the MG, the gradients of the loss function with respect to $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$ can be derived as shown below.

$$\frac{\partial \mathbf{L}}{\partial \mathbf{x}_{r1}^i} = \frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r1}^i} + \frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r1}^i} \odot \mathbf{G}_r^i + \frac{\partial \mathbf{L}}{\partial \mathbf{g}_r^i} \times \frac{\partial \mathbf{g}_r^i}{\partial \mathbf{x}_{r1}^i} \quad (5.8)$$

$$\frac{\partial \mathbf{L}}{\partial \mathbf{x}_{r2}^i} = \frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r2}^i} + \frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r2}^i} \odot \mathbf{G}_r^i + \frac{\partial \mathbf{L}}{\partial \mathbf{g}_r^i} \times \frac{\partial \mathbf{g}_r^i}{\partial \mathbf{x}_{r2}^i} \quad (5.9)$$

From the second terms of the equations (5.8) and (5.9), we can see that the gradients with respect to the ‘selected’ $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$ will be boosted by a quantity $\mathbf{G}_r^i$. As the loss function aims to decrease the distance between positive pairs and increase the distance between negative pairs, the explicit gating mechanism will also encourage the lower layers of the network to learn filters that can extract patterns that are more similar for positive pairs. For instance, if similarities are boosted and propagated for negative image pairs during training, the distance between the

negative pairs will be less and the loss of the network will be higher. This will be reflected in the gradients that is back-propagated from the loss layer. The third terms of the equations (5.8) and (5.9), shows the impact of the gate values ($\mathbf{g}_r$) on the loss function, i.e. the effect of a small change in the input features that affect the gate values which finally affects the loss function. Therefore, during back-propagation, the lower layers of the network will be encouraged to learn filters that can extract features ($\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$) that are more similar for positive pairs. Combined with the boosting mechanism of the MG, the overall gradients will be boosted and will encourage the lower layers to learn filters that can extract more similarities for positive pairs.

The key advantages of the proposed MG is that it is flexible in its architecture as well as differentiable. If the optimal variance factor $\mathbf{p}$ is learned to be high, it facilitates maximum information flow from the input to output and conversely if it is learned to be a low value, it allows only very similar patches to be boosted. The network learns to identify the optimal $\mathbf{p}$ for each dimension from the training data which results in a matching gate that is flexible in its functioning. Alongside learning an optimal $\mathbf{p}$, the network also learns the parameter $\mathbf{w}$ and the bias in Equation (5.1) to summarize the features along a horizontal stripe. Additionally, the MG can be inserted in between any layers or multiple layers in the network as it is a differentiable function. This will also facilitate end-to-end learning strategy in deep networks. The gradients for the MG with respect to its inputs $\mathbf{x}_{r1}$ and $\mathbf{x}_{r2}$ can be computed as follows.

Gradient Computation: The main equations for the computation of the gate values are given in Equations 5.1 to 5.5.

Let $\frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r1}}$ and $\frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r2}}$ be the gradients with respect to $\mathbf{a}_{r1}$ and $\mathbf{a}_{r2}$ respectively, back propagated from the loss layer of the proposed architecture. Below we show the derivation of the gradients with respect to each dimension of $\mathbf{y}_{r1}$ and the corresponding gradients for $\mathbf{y}_{r2}$ can be computed accordingly.

The derivative of the Loss ($\mathbf{L}$) with respect to $\mathbf{G}_r^i$ can be computed using the below expression.

$$\frac{\partial \mathbf{L}}{\partial \mathbf{G}_r^i} = \frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r1}^i} \odot \mathbf{x}_{r1}^i + \frac{\partial \mathbf{L}}{\partial \mathbf{a}_{r2}^i} \odot \mathbf{x}_{r2}^i \quad (5.10)$$

Once the gradient with respect to all the elements of $\mathbf{G}_r$ is computed, the gradients with respect to $\mathbf{g}_r$ can be computed as shown below.

$\frac{\partial \mathbf{L}}{\partial \mathbf{G}_r} \in \mathbb{R}^{1 \times c \times h}$ and $\frac{\partial \mathbf{L}}{\partial \mathbf{g}_r} \in \mathbb{R}^{1 \times 1 \times h}$. Therefore, the gradient with respect to the single element $\mathbf{g}_r^i$ can be computed as

$$\frac{\partial \mathbf{L}}{\partial \mathbf{g}_r^i} = \sum_{j=1}^c \frac{\partial \mathbf{L}}{\partial \mathbf{G}_r^{ji}} \quad (5.11)$$

The right hand side of the above equation (i.e. Equation (5.11)) denotes the sum along the dimension ‘ c ’ of $\frac{\partial \mathbf{L}}{\partial \mathbf{G}_r} \in \mathbb{R}^{1 \times c \times h}$ and $\frac{\partial \mathbf{L}}{\partial \mathbf{G}_r^i}$ indicate each dimension of $\frac{\partial \mathbf{L}}{\partial \mathbf{G}_r}$ along ‘ h ’.

Now,

$$\begin{aligned} \frac{\partial \mathbf{L}}{\partial \mathbf{y}_{r1}^i} &= \frac{\partial \mathbf{L}}{\partial \mathbf{g}_r^i} \times \frac{\partial \mathbf{g}_r^i}{\partial \mathbf{y}_{r1}^i} \\ &= \frac{\partial \mathbf{L}}{\partial \mathbf{g}_r^i} \times \exp\left(\frac{-(\mathbf{y}_{r1}^i - \mathbf{y}_{r2}^i)^2}{\mathbf{p}_i^2}\right) \times \frac{-2}{\mathbf{p}_i^2} \times (\mathbf{y}_{r1}^i - \mathbf{y}_{r2}^i) \end{aligned} \quad (5.12)$$

and

$$\frac{\partial \mathbf{L}}{\partial \mathbf{y}_{r2}^i} = -\frac{\partial \mathbf{L}}{\partial \mathbf{y}_{r1}^i} \quad (5.13)$$

Once the gradients $\frac{\partial \mathbf{L}}{\partial \mathbf{y}_{r1}^i}$ and $\frac{\partial \mathbf{L}}{\partial \mathbf{y}_{r2}^i}$ are obtained, the gradients with respect to $\mathbf{x}_{r1}^i$ and $\mathbf{x}_{r2}^i$ can be easily computed using the equations for the gradients of the PReLU as well as convolution layers.

Finally, the derivatives with respect to $\mathbf{p}_i$ can be computed using the following equation.

$$\begin{aligned} \frac{\partial \mathbf{L}}{\partial \mathbf{p}_i} &= \frac{\partial \mathbf{L}}{\partial \mathbf{g}_r^i} \times \frac{\partial \mathbf{g}_r^i}{\partial \mathbf{p}_i} \\ &= \frac{\partial \mathbf{L}}{\partial \mathbf{g}_r^i} \times \exp\left(\frac{-(\mathbf{y}_{r1}^i - \mathbf{y}_{r2}^i)^2}{\mathbf{p}_i^2}\right) \times (\mathbf{y}_{r1}^i - \mathbf{y}_{r2}^i)^2 \times \frac{2}{\mathbf{p}_i^3} \end{aligned} \quad (5.14)$$

5.2.1.3 Final Architecture:

The final architecture of the proposed system is shown in Figure 5.2. The baseline network is designed in such a way as to reduce the width of the feature map progressively without reducing the height from layers 4 – 6. This is essential to address the pose change of the human images across cameras while preserving the finer row-wise characteristics. As shown in the Figure 5.2, we inserted the proposed MG between the last 4 layers once the number of rows of the propagated feature maps is fixed.

5.2.2 Training and Optimization

5.2.2.1 Input preparation:

Siamese networks take image pairs as inputs. Therefore, we first pair all the images in the training set with a label ‘1’ indicating negative pairs and ‘0’ indicating the positive pairs following the previous work [143]. For large datasets, the number of negative image pairs will be orders of magnitude higher than the number of positive pairs. To alleviate this bias in the training set, we sample approximately 5 times the number of positive image pairs, as negative image pairs, for each subject. The mean image computed from all the training images is subtracted from all the images and the input pairs are fed to the network.

5.2.2.2 Training:

Both the baseline S-CNN model and the proposed architecture (Figure 5.2) are trained from scratch in an end-to-end manner with a batch size of 100 pairs in an iteration. The weight parameters (i.e. filters) of the networks are initialized uniformly following [93]. The gradients with respect to the feature vectors at the last layer are computed from the contrastive loss function and back-propagated to the lower layers of the network. Once all the gradients are computed at all the layers, we use mini batch stochastic gradient descent (SGD) to update the parameters of the network. Specifically, we use the adaptive per-parameter update strategy called the RMSProp [178] to update the weights. The decay parameter for RMSProp is fixed to 0.95 following previous works [126] and the margin for the contrastive loss function is kept as 1. Training is done for 20 epochs with an

early stopping strategy based on the saturation of the validation set performance. The initial learning rate is set to 0.002 and reduced by a factor of 0.9 after each epoch. The main hyper-parameter of the MG is the initial value of $\mathbf{p}$. We set this value to 4 initially and the network discovers the optimal value during learning. More details on parameter tuning and validation are given in Section 5.4

5.2.2.3 Testing:

During testing, each query image has to be paired with all the gallery images and passed to the network. The gating function can selectively boost the common patterns in each image pair. The Euclidean distance between the feature vectors obtained at the last layer is used to compare two input images. Once the distance between the query image and all the images in the gallery set are obtained, it is sorted in ascending order to find the top matches. The above procedure is done for all the query images and the final results are obtained. For an identity with multiple query images, the distances obtained for each query are rescaled in the range of 0 – 1 and then averaged.

5.3 Experiments

We provide a comprehensive evaluation of the proposed S-CNN architecture with the matching gate by comparing it against the baseline S-CNN architecture as well as other state-of-the-art algorithms for human re-identification. Majority of the systems for human re-identification are evaluated based on the Cumulative Matching Characteristics by treating human re-identification as a ranking problem. However, in [1], human re-identification is treated as a retrieval problem and the mean average precision (mAP) is also reported along with the Rank - 1 accuracy (R1 Acc). For a fair comparison, we report both mAP as well as the performance at different ranks for CUHK03 dataset and mAP and R1 Acc for Market-1501 dataset. For VIPeR dataset, we report only the CMC as it is the relevant measure [1]. All the implementations are done in MATLAB-R2015b and we use the MatConvNet package [184] for implementing all the proposed frameworks. Experiments were run on NVIDIA-Tesla K40 GPU and it took approximately 40-50 minutes per epoch on the CUHK03 dataset.

5.3.1 Datasets and settings

Experiments were conducted on challenging benchmark datasets such as, the Market-1501 [1] dataset, the CUHK03 [19] dataset and the VIPeR [16] dataset. The details of the datasets are given in Section 1.4. For Market-1501 dataset, we report the single-query (SQ) as well as the multi-query (MQ) evaluation results for this dataset. For multi-query evaluation, the matching scores obtained for each of the query images per identity are rescaled from 0-1 and averaged to obtain the final matching score. Following the standard evaluation protocols in [1], the Market-1501 dataset is split into 751 identities for training and 750 identities for testing. For CUHK03 dataset, the experiments presented in this chapter follow the ‘detected’ setting as this is closer to the real-world scenario. Following the splitting settings provided in [19], evaluation is conducted 20 times with 100 test subjects and the average result obtained at different ranks is reported. For multi-query evaluation, the matching scores obtained for each of the query images per identity are rescaled from 0-1 and averaged to obtain the final matching score. We also use 100 identities from the training set for cross-validation leaving out 1160 identities for training the network. The VIPeR dataset is relatively small and the number of distinct identities as well as positive pairs per identity for training are very less compared to the other datasets. Therefore, we conduct data augmentation as well as transfer learning from Market-1501 and CUHK03 datasets. For transfer learning, we remove the last fully connected layer in our baseline S-CNN architecture and then fine-tune the network using the VIPeR dataset. Removing the last fully connected layer was to avoid over-fitting by reducing the number of parameters. For the gated S-CNN framework, the MGs are inserted between layers 4 – 5 and 5 – 6. Other experimental settings are kept the same as in [138].

5.3.2 Results

The results for the Market-1501, CUHK03 and VIPeR datasets are given in Tables 5.2, 5.3 and 5.4 respectively. The proposed baseline S-CNN architecture outperforms majority of the existing approaches for human re-identification for Market-1501 and CUHK03 datasets at Rank 1. However, a few of the recent approaches such as [141, 167] outperforms ours for the CUHK03 dataset. [141] takes advantage of advanced Convolutional

Table 5.2: Performance Comparison of state-of-the-art algorithms for the Market-1501 dataset.

Method	Rank 1	mAP
SDALF [3] - CVPR 2010	20.53	8.20
eSDC [4] - CVPR 2013	33.54	13.54
BoW [1] - (SQ) - ICCV 2015	34.40	14.09
DNS [166] - (SQ) - CVPR 2016	61.02	35.68
Ours - Baseline - S-CNN - (SQ)	62.32	36.23
Ours - With Matching Gate - (SQ)	65.88	39.55
BoW [1] - (MQ) - ICCV 2015	42.14	19.20
BoW + HS [1] - (MQ) - ICCV 2015	47.25	21.88
DNS [166] - (MQ) - CVPR 2016	71.56	46.03
Ours - Baseline - S-CNN - (MQ)	72.92	45.39
Ours - With Matching Gate - (MQ)	76.04	48.45

Neural Network architecture, i.e. the GoogleNet while [167] proposes a regression framework which aims at scalable and fast human re-identification. For VIPeR dataset, even though our baseline S-CNN does not achieve the best results, it outperforms several other CNN based architectures [138, 140, 142]. Our final architecture with the MG improves over the baseline architecture by a margin of 4.2% and 1.6% at Rank 1 for CUHK03 and VIPeR datasets respectively. For Market-1501 dataset, our approach outperforms the baseline by a margin of 3.56% at Rank 1 for single query (SQ) setting and 3.12% at Rank 1 for multi query (MQ) setting.

For multi-camera networks, the mean average precision is a better measure for performance compared to the Rank - 1 accuracy [1] as it signifies how many of the correct matches are retrieved from various camera views. Therefore, compared to the improvement in Rank 1 accuracy, the mean average precision which indicates the retrieval accuracy may be more interesting for real-world applications with camera networks. Even though the mean average precision is not particularly important for CUHK03 dataset as it contains only two views, we report the mAP to compare the retrieval results of the proposed final architecture with the baseline S-CNN architecture. It can be seen that our final architecture with MG outperforms the mean average precision obtained by the baseline S-CNN by a margin of 3.32%, 3.06% and 3.27% for Market-1501-Single Query, Market-1501-Multi Query and CUHK03 datasets respectively.

Table 5.3: Performance Comparison of state-of-the-art algorithms for the CUHK03 dataset on the ‘detected’ setting.

Method	Rank 1	Rank 5	Rank 10	mAP
SDALF [3] - CVPR 2010	4.9	21.0	31.7	-
ITML [27] - ICML 2007	5.14	17.7	28.3	-
LMNN [28] - JMLR 2009	6.25	18.7	29.0	-
eSDC [4] - CVPR 2013	7.68	22.0	33.3	-
LDML [180] - ICCV 2009	10.9	32.3	46.7	-
KISSME [29] - CVPR 2012	11.7	33.3	48.0	-
FPNN [19] - CVPR 2014	19.9	49.3	64.7	-
BoW [1] - ICCV 2015	23.0	45.0	55.7	-
BoW + HS [1] - ICCV 2015	24.3	-	-	-
ConvNet [138] - CVPR 2015	45.0	75.3	55.0	-
LX [22] - CVPR 2015	46.3	78.9	88.6	-
MLAPG [30] - ICCV 2015	51.2	83.6	92.1	-
SS-SVM [164] - CVPR 2016	51.2	80.8	89.6	-
SI-CI [140] - CVPR 2016	52.2	84.3	92.3	-
DNS [166] - CVPR 2016	54.7	84.8	94.8	-
DGDropout [141] - CVPR 2016	75.3	-	-	-
HER [167] - BMVC 2016	65.2	92.2	96.8	-
Ours - Baseline - S-CNN	63.9	86.7	92.6	55.57
Ours - With Matching Gate	68.1	88.1	94.6	58.84

5.4 Analysis

In this section, we provide some analysis of the proposed baseline S-CNN as well as the Gated S-CNN.

5.4.1 Baseline

To evaluate the major design choices, we conducted experiments on the CUHK03 dataset with different variants of the baseline S-CNN. First, we conduct experiments by removing the batch normalization from the proposed baseline S-CNN architecture. Second, to evaluate the benefits of using smaller filters and a deeper architecture, we conducted experiments with larger filters and a less deeper architecture. Specifically, the architecture is $(7 \times 7, 5 \times 5, 5 \times 5, 5 \times 5, 4 \times 4, 9 \times 1)$ convolutions with 2×2 max-pooling in the first

Table 5.4: Performance Comparison of state-of-the-art algorithms using an individual method for the VIPeR dataset.

Method	Rank 1	Rank 5	Rank 10
LFDA [76] - CVPR 2013	24.1	51.2	67.1
eSDC [4] - CVPR 2013	26.9	47.5	62.3
Mid-level [137] - CVPR 2014	29.1	52.3	65.9
SVMML [5] - CVPR 2013	29.4	63.3	76.3
VWCM [26] - ECCVW 2014	30.7	63.0	76.0
SalMatch [31] - ICCV 2013	30.2	52.3	65.5
QAF [160] - CVPR 2015	30.2	51.6	62.4
SCNN [142] - ICPR 2014	28.2	59.3	73.5
ConvNet [138] - CVPR 2015	34.8	63.7	75.8
CMWCE [23] - ICPR 2014	37.6	68.1	81.3
SCNCD [24] - ECCV 2014	37.8	68.5	81.2
LX [22] - CVPR 2015	40.0	68.1	80.5
PRCSL [161] - ICCV 2015	34.8	68.7	82.3
MLAPG [30] - ICCV 2015	40.7	69.9	82.3
MT-LORAE [162] - ICCV 2015	42.3	72.2	81.6
Semantic Representation [163] - CVPR 2015	41.6	71.9	86.2
DGDropout [141] - CVPR 2016	38.6	-	-
SI-CI [140] - CVPR 2016	35.8	67.4	83.5
SS-SVM [164] - CVPR 2016	42.7	-	84.3
MCP-CNN [139] - CVPR 2016	47.8	74.7	84.8
HGD [165] - CVPR 2016	49.7	79.7	88.7
DNS [166] - CVPR 2016	51.7	82.1	90.5
HER [167] - BMVC 2016	53.0	79.8	89.6
SCSP [168] - CVPR 2016	53.5	82.6	91.5
Ours - Baseline - S-CNN	36.2	65.1	76.3
Ours - With Matching Gate	37.8	66.9	77.4

three layers (Conv Blocks). Finally, To evaluate the benefits of using asymmetric filters, we conduct experiments using symmetric filters. Specifically, we use 4×4 and 3×3 instead of 1×4 and 1×3 respectively. All the results are indicated in Table 5.5. It can be seen that batch-normalization is advantageous and the choice of smaller asymmetric

filters can benefit the performance of the human re-identification system.

Table 5.5: Baseline variants.

Method	Rank 1	Rank 5	Rank 10
Without BatchNorm	57.6	83.3	89.8
Large Filters	61.4	85.1	91.1
Symmetric filters	62.1	85.6	92.0
Ours	63.9	86.7	92.6

5.4.2 Gated S-CNN

In this section, we give an analysis of the baseline architecture. First, the feature summarization strategy is evaluated. Then the initial value of the variance factor ‘ $\mathbf{p}$ ’ is evaluated and finally, we evaluate the performance by inserting the matching gate at multiple layers.

5.4.2.1 Gate Visualizations

The visualization of the gating mechanism in the proposed matching gate is shown in Figure 5.3. Figure 5.3 (a) shows a query image and a hard negative image (example shown in Figure 5.1 (b)). The middle row shows the average feature activations at the output of the 4th convolutional block which is the input to the proposed gating function and the third row shows the obtained gate values using the proposed gating function. It can be seen that for the first few rows where the subject in the query is wearing a hat, the gate activations are low indicating lower similarity where as for a few middle rows, the gate activations are high indicating higher similarity. In Figure 5.3 (b), we show the image paired with its true positive, the layer 5 inputs and the gate values. It can be seen that for majority of the patches, the gate values are high indicating high similarity between the image patches. This indicates that the gating function can efficiently extract relevant common information from the feature maps of both the images and boost them.

Figure 5.3: Gate Visualization: (a) Query paired with its hard-negative (b) Query paired with its positive. The layer 5 input values as well as the corresponding gate values obtained for both pairs are also shown. Boxes of same color indicates corresponding regions in the images.

5.4.2.2 Feature Summarization strategy:

In the proposed feature summarization strategy, the features along a row are aggregated and compared against the corresponding aggregated features along the same row in the paired image. Instead of comparing along each row, we conducted experiments by comparing the feature maps of the whole image in the MG. We use filters of size $\mathbf{w} \in \mathbb{R}^{R \times c \times h \times h}$ where R is the total number of rows. Results are given in Table 5.6. The obtained results are quite close to those obtained by the baseline (slightly inferior). We believe that this is because, matching over the whole image at early/middle layers results in a large number of parameters as the feature maps are very large. Additionally, the approach fails to identify local matching patterns as the gating is done based on the global similarity. This results in a performance similar to the baseline.

Another strategy is to use filters of size $\mathbf{w} \in \mathbb{R}^{1 \times c}$ repeated h times instead of $\mathbf{w} \in$

Table 5.6: Different feature summarization strategies.

Method	Rank 1	Rank 5	Rank 10
Match over whole image	63.3	86.4	92.1
$1 \times c$ repeated h times	64.9	87.0	92.8
Our baseline S-CNN	63.9	86.7	92.6
Our Gated S-CNN	68.1	88.1	94.6

$\mathbb{R}^{1 \times c \times h \times h}$. A filter ($\mathbf{w}$) of size $\mathbf{w} \in \mathbb{R}^{1 \times c \times h \times h}$ is essentially a fully connected layer that can also learn interactions between the feature maps (i.e. along the depth) in a row. We conducted experiments with a filter of size $\mathbf{w} \in \mathbb{R}^{1 \times c}$ repeated h times. The results are indicated in Table 5.6. It can be observed that the results are inferior to the results obtained by the proposed feature summarization strategy. We believe that this is because, filter of size $\mathbf{w} \in \mathbb{R}^{1 \times c}$ repeated h times lacks the capability to learn the interactions between different feature maps (along the depth) in a layer. However, it is also noteworthy that the performance is better than the baseline indicating the benefits of the gating technique.

5.4.2.3 Initial value for variance factor ($\mathbf{p}$):

We first conducted validation experiments to tune the initial value for the variance factor $\mathbf{p}$ for the matching gate. The values for $\mathbf{p}$ was sampled in a log-uniform range of $[2^{-4}, \dots, 2^5]$. The validation performance at Rank 1 as well as the mean average precision (mAP) is shown in Figure 5.4 (a). As mentioned in the main manuscript, a higher initial value for $\mathbf{p}$ ensures smooth flow of gradients through the matching gate function. Even though the matching gate may learn an optimal value with a smaller initial value, we empirically found that the value 4 can achieve the best validation results at Rank 1. Therefore, the initial value for $\mathbf{p}$ was kept as 4 in all our experiments for all the matching gates in our final architecture.

5.4.2.4 Adding the Matching Gate at multiple layers:

To analyze the effect of adding the matching gate between different layers, experiments were conducted by inserting the matching gate between multiple layers starting from the

last layer as well as the first layer . We also give the validation performance without any matching gate inserted into the architecture which will result in our baseline S-CNN architecture. The validation curves are shown in Figure 5.4 (b). The X-axis shows the number of matching gates inserted starting from the first or last layer. The solid curve shows the Rank 1 accuracy (Red color) and the mean average precision (blue color) for the results of the matching gate inserted from the last layer. i.e. X-value = 1 indicates that the matching gate is inserted only between Conv Blocks 6 and 7 and X-value = 2 indicates matching gates are inserted between Conv Blocks 6 and 7 as well as 5 and 6 and so on. The dotted line indicates that the matching gates are inserted starting from the first layer. i.e. X-value = 1 indicates that the matching gate is inserted only between Conv Blocks 1 and 2 and X-value = 2 indicates matching gates are inserted between Conv Blocks 1 and 2 as well as 2 and 3 and so on. The matching gates are inserted after the pooling layers for the first three conv blocks to avoid over-fitting as the parameters learned must be large if they are to be inserted before the pooling layer. For the curve that shows the matching gate inserted from the last layers, it can be seen from Figure 5.4 (b) that the performance increases steadily as the matching gate is inserted at multiple layers, but beyond the last 3 layers, the Rank 1 accuracy does not show much gain in the validation performance. Similarly, for the case when the matching gate is inserted from the first layers, we observe that in the initial few layers, the performance does not improve much, but as the matching gate is inserted towards towards the last layers the gain is comparatively higher. Additionally, the complexity also increases as the gate is inserted between the early layers. For instance, for the matching gate inserted between layers 1 and 2, the feature summarization unit requires convolutions with filters of size $\mathbb{R}^{1 \times 32 \times 32 \times 32}$ for a feature map of size $\mathbb{R}^{64 \times 32 \times 32}$. Therefore, even though there is a slight gain in performance, we limit the insertion of the matching gates between the last three layers only. By comparing the effect of adding matching gates starting from the first layer as well as the last layer, we observed that adding the MG from last layer is more useful. This could be because, at the initial layers, the network extract edges and a combination of edges which are less informative compared to the mid-level and high level features which encode more abstract concepts. Therefore, intuitively, it is more meaningful to insert the matching gate towards the final layers than starting from the first layers. From

our validation curves in Figure 5.4 (b), we observed that inserting the MG between the last three layers (solid red curve and the solid blue curve in Figure 5.4 (b)) is the best choice as the performance is observed to be almost stable for inserting the MG beyond the last three layers.

5.5 Conclusion and Future Works

In this chapter, we have proposed a baseline siamese CNN and a learnable Matching Gate function for siamese CNN that can vary the network behavior during training and testing for the task of human re-identification. The Matching Gate can compare the local features along a horizontal stripe for an input image pair during run-time and adaptively boost local features for enhancing the discriminative capability of the propagated features. The gating function is also designed to be a differentiable one with learnable parameters for adjusting the variance of the gate values as well as for summarizing the horizontal stripe features. This is essential for adjusting the amount of filtering at each stage of the network as well as to facilitate end-to-end learning of deep networks. We have conducted experiments on the Market-1501 dataset, the CUHK03 dataset and the VIPeR dataset to evaluate how run-time feature selection can enable the network to learn more discriminative features for extracting meaningful similarity information for an input pair. The introduction of the gating function in between convolutional layers results in significant improvement of performance over the baseline S-CNN. Our S-CNN model with the matching gate achieves promising results compared to the state-of-the-art algorithms on the above datasets. Partial works presented in this chapter has been published in [185].

(a)

(b)

Figure 5.4: Validation Curves: (a) The validation performance vs the initial values for p . (b) The validation performance vs the number of layers the matching gate is inserted between, starting from the last as well as the first layers.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

This thesis focuses on studying the feature learning frameworks for human re-identification, specifically, learning invariant features. Several approaches for dealing with the major challenges in human re-identification such as variations in illumination, pose and view-point are introduced and detailed analysis and experiments were conducted to evaluate the effectiveness of the proposed approaches.

First, a framework is proposed to learn invariant color features. Since the human re-identification problem is characterized by several challenges as mentioned above, learning invariant features is essential for achieving superior re-identification rate. Based on the observations that color features are the most important cue for re-identification, we first propose an invariant color feature learning framework. Since the perceived color of the subject may vary due to variations in illumination across camera views, we propose a framework to learn a mapping and encoding scheme to project raw pixels to a color constant space. The intuition behind the proposed approach is that, in the encoded space, the representations of similar colors must be closer to each other. Experiments were conducted on synthetic color constancy datasets as well as human re-identification datasets to prove the effectiveness of our approach. To test the generalization capabilities of the feature learning and encoding framework, the proposed framework was also applied to an image retrieval problem based on color features. Combining the invariant color features with other types of learned texture features, promising results were demonstrated on several benchmark datasets.

Second, to strengthen the discriminative capability of the invariant feature learning framework, contextual information is also incorporated to the local features. Conventional features are extracted at a small local region and are independent of the features from other regions. This may result in failing to capture the contextual information, i.e. information from other regions of the image. Recurrent Neural Networks are capable of capturing the dependency between sequential features. In the proposed approach, the image is treated as a sequence of regions and a Siamese Long Short Term Memory (LSTM) Recurrent Neural Network architecture is proposed to capture the spatial dependencies between different parts of the pedestrian image. In addition to the spatial dependency, the sophisticated gating mechanisms in the LSTM cells can capture relevant contextual information and propagate them to the rest of the network. Selecting the relevant contextual information is essential as the features are learned from raw-pixel values which may contain background and other noisy information. Experiments were conducted on standard human re-identification benchmarks and promising results were achieved comparing to the state-of-the-art approaches.

Third, an end-to-end Gated Siamese Convolutional Neural Network (Gated S-CNN) architecture is proposed to model the pairwise information between the images for human re-identification. A novel matching gate (MG) is proposed in this work to selectively boost the common information between image pairs. The intuition is that, the subtle patterns which are essential to distinguish between a positive pair and a hard-negative pair may not be given enough emphasis in the middle layers of the network. The proposed MG can identify such common patterns from the middle layers and boost them so that those information are available at the final layer embedding. Thus, the feature vectors are adaptive to the input pairs, thus making it matching-aware from the middle layers. The proposed matching gate can be inserted in between any layers of the network as well as it is modeled as a differentiable module to facilitate end-to-end learning. Experiments were conducted on large benchmark datasets for human re-identification and the efficiency of the proposed approach was proven compared to a baseline Siamese Convolutional Neural Network architecture. Further experiments were conducted to compare the proposed approach with state-of-the-art algorithms and the Gated S-CNN achieves promising results on these benchmarks.

6.2 Future Works

To successfully solve the human re-identification task and to apply it in a real-world scenario, there are many more challenges that need to be addressed. In addition to looking at the problem from a real-world application perspective, the techniques introduced in this thesis also provides enormous possibilities for future research. To conclude the thesis, first a comparative study of the approaches presented in thesis is presented and further, some promising directions for future research are proposed.

Varying illumination conditions across different camera views have been a major challenge in human re-identification research. The proposed invariant color feature learning framework addresses this problem by proposing a learning framework to project pixels belonging to similar color to be close to each other. Though its effectiveness is evident from the experimental results, it cannot be considered a scalable approach as it requires expensive manual annotation of patches from images captured by multiple cameras in addition to the image annotation itself. It is also desirable to use a holistic approach to learning invariant features rather than addressing each specific challenge individually, such as varying illumination. To this end, the Siamese LSTM as well as the Gated S-CNN approaches seem to be more practical compared to the invariant color feature learning framework. These approaches rely on the image level annotation alone for optimizing the framework. The LSTM framework aims at modeling the spatial context to improve the discriminative capability of the local features and the Gated S-CNN framework aims at amplifying the subtle local similarity cues for distinguishing positive pairs from hard-negative pairs. These two frameworks approach the human re-identification problem from different perspectives and hence can be considered complementary to each other. However, currently the proposed LSTM framework relies on hand-crafted features which limits its potential to learn features from raw data, i.e. the image itself. However, in the Gated S-CNN framework, this drawback is eliminated. The re-identification performance on large datasets such as Market-1501 and CUHK03 datasets clearly indicate the advantage of the end-to-end trained S-CNN framework over the LSTM framework which relies on hand-crafted approaches. It also has the advantage of incorporating the information from the paired input image while learning the representation for an image. However, the Gated S-CNN framework does not incorporate any spatial context cues from the input

image which is a complementary aspect considered in the LSTM framework. Though in terms of human re-identification performance, Gated S-CNN achieves the best results on large datasets, the LSTM and the S-CNN framework can be considered complementary and can benefit from the advantage of each other.

The aforementioned limitations of the proposed frameworks give scope for numerous possibilities for future research. For the invariant color feature learning framework introduced in Chapter 3, the proposed model relies on approximate correction algorithms to correct the pixel values initially. However, it is desirable to eliminate the dependency on hand-crafted approaches as they are based on several assumptions about the statistics of color distribution in the images. Additionally, different illuminations require different transformations to project them back to the canonical illuminant space. Hence, learning multiple transformations for modeling different illumination variations can be considered as a possible direction for future research. The model has to be designed in a data-driven manner to learn the corresponding transformations from the pixels that are corrupted by illumination.

For the feature selection techniques such as the LSTM based contextual dependency modeling presented in Chapter 4 and Gated Siamese Convolutional Neural Network approach presented in Chapter 5, enormous possibilities for future directions exist. A straightforward way is to combine both the approaches to build a unified model that can learn features end-to-end for human re-identification. The LSTM cells in the network can model the spatial dependency while the matching gate can model the pairwise dependency of the images. Thus the advantages are two fold. (1) The dependency of hand-crafted features can be eliminated for the Siamese LSTM network. (2) The spatial dependency can be incorporated to the Gated S-CNN architecture. Another direction to extend the Gated S-CNN architecture is by exploring whether dynamically weighting the similarities can be more beneficial compared to fixed weights, i.e whether weight coefficients can be helpful while boosting the similarities across images. This will enable the network to control the amount of boosting done for negative pairs and also gives flexibility to select ‘what regions to be boosted more than the other’. Other possibilities include exploring whether deeper networks such as the Inception [79] or the deep residual nets [80] can perform better for human re-identification.

In a general application perspective, several possibilities exist for future works. The real-world re-identification task is very challenging in nature and enormous problems are yet to be explored. First, the re-identification systems are tested in mostly a forensic manner. For real-time re-identification, it is necessary to consider the time gap for the subject to move from one camera to the other. Additionally, the subject can take multiple paths for reaching one location from the other. Hence, it is also necessary to consider which cameras to be activated to continuously search the pedestrians. Besides these challenges, there are many more problems that can be explored for both real-time as well as forensic applications. One major challenge is re-identification in crowded scenes. Environments such as railway station, busy markets or shopping malls may be very crowded and re-identification in such environments are very challenging since pedestrians may be occluded each other or by the presence of other objects. Therefore, techniques to identify the most discriminative parts from the observable human body is essential. Scalability is another issue that is prominent with re-identification systems. Since public places are equipped with thousands of cameras, identifying an individual from these cameras can be extremely challenging. Even though treating human re-identification as a retrieval problem can remove the dependency on pairwise camera based models, it is still a challenging task as the search space is enormous and many pedestrians with similar appearance can be found from different cameras. Other challenges include human re-identification in factory or school environments where the clothing is uniform for all the subjects. In such scenarios, it becomes increasingly difficult to distinguish subjects based on their appearance. In such scenarios, other cues such as more intricate patterns based on appearance such as the physical characteristics of the body or skin color may have to be used. A new application or a new human re-identification scenario is description based person search, i.e. to find an individual based on the natural language description of the subject's appearance. This can help to eliminate the dependency of human intervention in the re-identification loop. However, I believe that the proposed solutions to the re-identification problems can be a promising step as a basis for the aforementioned future directions.

Appendix

Color Feature Learning Framework applied to Image Retrieval Task

The invariant color feature learning and encoding scheme for the pixels proposed in Chapter 3 have been demonstrated to be effective for the task of learning illumination invariant color based features. To test the generalization capabilities of the proposed feature learning and encoding approach, experiments were conducted on an Image Retrieval task. In the image retrieval setting, the objective is to retrieve matching images from a set of gallery images for a given query image.

Image Retrieval

The proposed framework without the invariance condition is applied to image retrieval task. The objective is to test the robustness and the generalization capabilities of the auto-encoder based learning scheme and sparse coding based encoding scheme. Hence we avoid the invariance condition for the above application. The details regarding the framework is given below.

Framework

In the proposed framework, the training data are 3-dimensional pixel values. We randomly sample 200k pixels from the entire dataset. For each image, a histogram equalization is performed for the R, G and B channels before sampling the pixels. Such randomly sampled pixels are standard normalized and thus the input data is prepared for the system.

Objective formulation and optimization: The objective of the system is to capture the stable structures and patterns inherent in the data. We extract such information from the pixels based on the reconstruction error. This concept is close to the patch based auto-encoder. The 3-dimensional pixel values are transformed to a higher dimensional space and reconstructed to the original 3-dimensional space. The objective is to minimize the reconstruction error for all the pixel values. Mathematically, it can be written as,

$$\begin{aligned} \underset{\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2}{\text{minimize}} \quad & \frac{1}{m} \sum_{i=1}^m \|(W_2'(W_1'x_i + b_1) + b_2) - x_i\|_2^2 \\ & + \lambda(\|\mathbf{W}_1\|_F^2 + \|\mathbf{W}_2\|_F^2) \end{aligned} \quad (1)$$

where $\|\cdot\|_F$ denotes the Frobenius norm. $x_i \in \mathbb{R}^{3 \times 1}$ is the i^{th} pixel sampled randomly from the images and i ranges from 1 to m where m is the total number of pixels sampled. $W_1 \in \mathbb{R}^{3 \times h}$, $W_2 \in \mathbb{R}^{h \times 3}$, $b_1 \in \mathbb{R}^{h \times 1}$, $b_2 \in \mathbb{R}^{3 \times 1}$ are the auto-encoder parameters. h is the dimension of the projected space. All the parameters are randomly initialized and a gradient based optimization is done for the above objective function. Gradients of the objective function in equation 1 with respect to each term are computed and we use L-BFGS optimization scheme to find the local optima. Let L be the objective function in equation 1. The partial derivatives of L with respect to each parameter are as shown below.

$$\frac{\partial L}{\partial W_1} = 2\lambda W_1 + \frac{2}{m} \sum_{i=1}^m ((W_2'(W_1'x_i + b_1) + b_2) - x_i) \times x_i' \times W_2' \quad (2)$$

$$\frac{\partial L}{\partial W_2} = 2\lambda W_2 + \frac{2}{m} \sum_{i=1}^m (W_1'x_i + b_1) \times ((W_2'(W_1'x_i + b_1) + b_2) - x_i)' \quad (3)$$

$$\frac{\partial L}{\partial b_1} = \frac{2}{m} \sum_{i=1}^m W_2 \times ((W_2'(W_1'x_i + b_1) + b_2) - x_i) \quad (4)$$

$$\frac{\partial L}{\partial b_2} = \frac{2}{m} \sum_{i=1}^m ((W_2'(W_1'x_i + b_1) + b_2) - x_i) \quad (5)$$

After learning the auto-encoder parameters, the inputs are transformed into the higher dimensional space by the following transformation.

$$y_i = (W_1'x_i + b_1) \quad (6)$$

After $y_i \in \mathbb{R}^{h \times 1}$ is obtained for all the training samples x_i , we compute a dictionary D to encode each y_i as sparse codes. The objective for sparse coding can be written mathematically as shown below.

$$\underset{\mathbf{D}}{\text{minimize}} \quad \frac{1}{m} \sum_{i=1}^m \|y_i - D\alpha_i\|_2^2 + \gamma \|\alpha_i\|_1 \quad (7)$$

where $D \in \mathbb{R}^{h \times d}$ are the basis vectors (Dictionary or Codebook) to encode each of the transformed pixel values in the higher dimensional space and d is the number of such learned dictionary elements. $\alpha_i \in \mathbb{R}^{d \times 1}$ is the sparse code generated for y_i based on the dictionary D . $\|\alpha_i\|_1$ enforces sparsity and γ is the penalty for the sparsity term. We use the SPAMS toolbox [154] to learn the dictionary. The proposed architecture has only one layer and the main parameters of our system are $h = 60$, $d = 250$, $\lambda = 3 \times 10^{-3}$ and $\gamma = 0.1$ as used in [172]. The time required for learning the transformation and dictionary are 3.342s and 27.20s respectively on a Core i5 3.20 GHz PC.

Feature computation: After learning the auto-encoder parameters and the dictionary, features are generated for each image. The images are resized to 256×256 and a histogram equalization is performed. The pixels are standard normalized and transformed to the higher dimensional space by following the equation 1. In the higher dimensional space, each of the pixels are encoded using the dictionary D following equation 7. Thus the original $256 \times 256 \times 3$ dimensional image becomes $256 \times 256 \times d$ dimensional. Further a max pooling scheme is adopted for the encoded image over 32×32 , 64×64 , 96×96 , 112×112 and 128×128 regions. Due to the presence of multiple pooling operations at different sizes, the average time taken to compute the feature vector for each image is nearly 8.2s. The final 23000 dimensional feature vector can be obtained by concatenating these pooled features over all the regions in the image.

Distance computation: For computing the distance between the feature vectors of each image, we follow [186] and use the L_1 distance. We also computed the L_2 distance,

L_∞ and χ^2 distance, but observed that L_1 distance gave the best results. For each image, the distance between all the images are computed and the matching scores are obtained. These matching scores are used for evaluation. More details are given in the experiments section.

Experiments

We evaluate our approach on INRIA Holidays [187] dataset and the UKB dataset [188]. The results are reported as the mean average precision (mAP). Histograms in different color spaces such as RGB (RGBHist), HSV (HSVHist), YUV (YUVHist) and Lab (LabHist) are computed on 32×32 , 64×64 , 96×96 , 112×112 and 128×128 regions for an image and the final feature vector is the concatenation of the features computed for these regions. We compare our approach with these baselines (histograms) as well as other standard approaches and the results are as reported in the corresponding tables.

INRIA Holidays dataset: The dataset consists of 1491 photographs of natural sceneries and vacation spots. There are 500 groups of images based on same scene or object. From each group, one image serves as query and the retrieval is done based on this query. The performance is reported as mAP over these 500 queries. Some of the images in the dataset are rotated by ± 90 degrees and are not in their natural orientation. We manually bring all images in the dataset to the normal orientation as done in [189]. The results are reported for both the rotated and unrotated version of this dataset. We observed an improvement of 3.186 mAP for the rotated dataset.

As shown in the table 1, the proposed encoding scheme outperforms the baselines such as the histograms in several standard color spaces. In table 2, we show the comparison of our approach with other approaches. It can be seen that our approach outperforms many standard approaches. The Hamming Embedding (HE) approaches in [190] and [186] which outperforms our method for this dataset utilizes the combination of color and texture features. But it should be noted that, the proposed color based signature performs better in all the other cases and also outperforms several method that works based on a combination of color and texture features.

UKB dataset: The dataset includes 10,200 indoor photographs of 2550 objects (4 photos per object). Each image is used to query the rest of the dataset. The performance

Table 1: Performance Comparison of the proposed LCF with different baseline color feature descriptors on the INRIA Holidays and UKB Dataset.

Method	mAP - Holidays	mAP - UKB
RGBHist	48.198	2.51
HSVHist	48.580	2.29
YUVHist	36.378	2.40
LabHist	34.097	2.14
LCF	66.324	3.03

Table 2: Performance Comparison of the LCF with different color based approaches based on the INRIA Holidays Dataset

Method	Feature Descr.	D	Holidays
BOF [190]	SIFT	20000	46.9
BOF [190]	SIFT	200000	57.2
HE [190]	SIFT	N/A	74.5
colorGIST [191]	RGB Pixels	960	37.6
BOF [186]	colorSIFT	20000	57.9
HE [186]	colorSIFT	N/A	76.5
BOC [186]	Lab Pixels	1024	64.6
BOC [186]	Lab Pixels	256	63.8
LCF	RGB Pixels	23000	66.324

is reported as the average number of same-object images within the top-4 results, and is a number between 0 and 4. Table 1 shows the comparison of our approach against the baseline histograms and it can be seen that our approach outperforms all the baseline color features. However, from table 3, it can be seen that among other color based features, BOC [186] works better than our approach. HE [190] which outperform ours work based on a combination of color and texture. However, it should be noted that our method with color descriptors alone can outperform the results of HE [186] which is a combination of color and texture signatures.

Table 3: Performance Comparison of the LCF with different color based approaches based on the UKB Dataset

Method	Feature Descr.	D	UKB
BOF [190]	SIFT	20000	2.88
BOF [190]	SIFT	200000	2.95
HE [190]	SIFT	N/A	3.30
colorGIST [191]	RGB Pixels	960	2.06
BOF [186]	colorSIFT	20000	2.28
HE [186]	colorSIFT	N/A	2.58
BOC [186]	Lab Pixels	256	3.34
LCF	RGB Pixels	23000	3.03

References

- [1] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, J. Bu, and Q. Tian, “Scalable person re-identification: A benchmark,” in *IEEE International Conference on Computer Vision (ICCV)*, 2015.
- [2] R. Layne, T. M. Hospedales, and S. Gong, “Domain transfer for person re-identification,” in *Proceedings of the ACM/IEEE International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Stream (ARTEMIS)*, 2013.
- [3] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, “Person re-identification by symmetry-driven accumulation of local features,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.
- [4] R. Zhao, W. Ouyang, and X. Wang, “Unsupervised salience learning for person re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [5] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, “Learning locally-adaptive decision functions for person verification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [6] I. Kviatkovsky, A. Adam, and E. Rivlin, “Color invariants for person reidentification,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2013.
- [7] B. Ma, Y. Su, and F. Jurie, “Bicov: a novel image representation for person re-identification and face verification,” in *Proceedings of the British Machine Vision Conference (BMVC)*, 2012.

REFERENCES

- [8] C. Liu, S. Gong, C. Loy, and X. Lin, “Person re-identification: What features are important?,” in *European Conference on Computer Vision, Workshops and Demonstrations (ECCVW)*, 2012.
- [9] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino, “Custom pictorial structures for re-identification,” in *Proceedings of the British Machine Vision Conference (BMVC)*, BMVA Press, 2011.
- [10] D. Gray and H. Tao, “Viewpoint invariant pedestrian recognition with an ensemble of localized features,” in *European Conference on Computer Vision (ECCV)*, 2008.
- [11] J. M. Morel, A. B. Petro, and C. Sbert, “Fast implementation of color constancy algorithms,” *Proceedings of the SPIE*, 2009.
- [12] B. Shuai, Z. Zuo, G. Wang, and B. Wang, “Dag-recurrent neural networks for scene labeling,” *CoRR*, vol. abs/1509.00552, 2015.
- [13] Z. Zuo, B. Shuai, G. Wang, X. Liu, X. Wang, B. Wang, and Y. Chen, “Convolutional recurrent neural networks: Learning spatial dependencies for image representation,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2015.
- [14] Z. Zuo, B. Shuai, G. Wang, X. Liu, X. Wang, B. Wang, and Y. Chen, “Learning contextual dependence with convolutional hierarchical recurrent neural networks,” *IEEE Transactions on Image Processing (TIP)*, 2016.
- [15] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, “Signature verification using a “siamese” time delay neural network,” in *Advances in Neural Information Processing Systems (NIPS)*, 1994.
- [16] D. Gray, S. Brennan, and H. Tao, “Evaluating appearance models for recognition, reacquisition, and tracking,” in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, 2007.
- [17] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof, “Person re-identification by descriptive and discriminative classification,” in *Proceedings of the Scandinavian Conference on Image Analysis (SCIA)*, 2011.

REFERENCES

- [18] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino, “Custom pictorial structures for re-identification,” in *Proceedings of the British Machine Vision Conference (BMVC)*, 2011.
- [19] W. Li, R. Zhao, T. Xiao, and X. Wang, “Deepreid: Deep filter pairing neural network for person re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [20] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2010.
- [21] H. Moon and P. J. Phillips, “Computational and performance aspects of PCA-based face-recognition algorithms,” *Perception-London*, 2001.
- [22] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, “Person re-identification by local maximal occurrence representation and metric learning,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [23] Y. Yang, S. Liao, Z. Lei, D. Yi, and S. Z. Li, “Color models and weighted covariance estimation for person re-identification,” *Proceedings of International Conference on Pattern Recognition (ICPR)*, 2014.
- [24] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li, “Salient color names for person re-identification,” in *European Conference on Computer Vision (ECCV)*, 2014.
- [25] X. Wang and R. Zhao, “Person re-identification: System design and evaluation overview,” in *Person Re-Identification*, 2014.
- [26] Z. Zhang, Y. Chen, and V. Saligrama, “A novel visual word co-occurrence model for person re-identification,” in *European Conference on Computer Vision Workshop on Visual Surveillance and Re-Identification (ECCV Workshop)*, 2014.
- [27] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, “Information-theoretic metric learning,” in *Proceedings of the International Conference on Machine Learning (ICML)*, 2007.

REFERENCES

- [28] K. Q. Weinberger and L. K. Saul, “Distance metric learning for large margin nearest neighbor classification,” *The Journal of Machine Learning Research (JMLR)*, 2009.
- [29] M. Kostinger, M. Hirzer, P. Wohlhart, P. Roth, and H. Bischof, “Large scale metric learning from equivalence constraints,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [30] 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 (ICCV)*, 2015.
- [31] R. Zhao, W. Ouyang, and X. Wang, “Person re-identification by salience matching,” in *IEEE International Conference on Computer Vision (ICCV)*, 2013.
- [32] W. Li, R. Zhao, and X. Wang, “Human reidentification with transferred metric learning,” in *Proceedings of Asian Conference on Computer Vision (ACCV)*, 2012.
- [33] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, “Local fisher discriminant analysis for pedestrian re-identification,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [34] F. Xiong, M. Gou, O. Camps, and M. Sznai, “Person re-identification using kernel-based metric learning methods,” in *European Conference on Computer Vision (ECCV)*, 2014.
- [35] C. Su, F. Yang, S. Zhang, Q. Tian, L. S. Davis, and W. Gao, “Multi-task learning with low rank attribute embedding for person re-identification,” in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [36] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2002.
- [37] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision (IJCV)*, 2004.

- [38] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikainen, and S. Li, “Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.
- [39] A. Albiol, A. Albiol, J. Oliver, and J. M. Mossi, “Who is who at different cameras: people re-identification using depth cameras,” *IET Computer Vision*, 2012.
- [40] R. Satta, F. Pala, G. Fumera, and F. Roli, “Real-time appearance-based person re-identification over multiple kinect tm cameras,” in *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2013.
- [41] I. B. Barbosa, M. Cristani, A. Del Bue, L. Bazzani, and V. Murino, *Re-identification with RGB-D Sensors*. 2012.
- [42] F. Pala, R. Satta, G. Fumera, and F. Roli, “Multimodal person reidentification using rgb-d cameras,” *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 2016.
- [43] E. H. Land and J. J. McCann, “Lightness and retinex theory,” *Journal of the Optical Society of America*, 1971.
- [44] E. H. Land, *The retinex theory of color vision*. Scientific America., 1977.
- [45] G. Buchsbaum, “A spatial processor model for object colour perception,” *Journal of the Franklin Institute*, 1980.
- [46] G. D. Finlayson and E. Trezzi, “Shades of gray and colour constancy,” in *Twelfth Color Imaging Conference: Color Science and Engineering Systems, Technologies, and Applications*, 2004.
- [47] J. van de Weijer, T. Gevers, and A. Gijsenij, “Edge-based color constancy,” *IEEE Transactions on Image Processing (T-IP)*, 2007.
- [48] D. Forsyth, “A novel algorithm for color constancy,” *International Journal of Computer Vision (IJCV)*, 1990.

REFERENCES

- [49] G. Finlayson, S. Hordley, and P. Hubel, “Color by correlation: a simple, unifying framework for color constancy,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2001.
- [50] G. Finlayson, S. Hordley, and I. Tastl, “Gamut constrained illuminant estimation,” *International Journal of Computer Vision (IJCV)*, 2006.
- [51] A. Gijsenij, T. Gevers, and J. van de Weijer, “Generalized gamut mapping using image derivative structures for color constancy,” *International Journal of Computer Vision (IJCV)*, 2010.
- [52] D. H. Brainard and W. T. Freeman, “Bayesian color constancy,” *Journal of Optical Society of America A*, 1997.
- [53] M. Ebner, “Evolving color constancy,” *Pattern Recognition Letters*, 2006.
- [54] A. Gijsenij, T. Gevers, and J. van de Weijer, “Computational color constancy: Survey and experiments,” *IEEE Transactions on Image Processing (TIP)*, 2011.
- [55] M. Ebner, *Color Constancy*. The Wiley-IS&T Series in Imaging Science and Technology, 2007.
- [56] S. D. Hordley, “Scene illuminant estimation: past, present, and future,” *Color Research & Application*, 2006.
- [57] D. Berwick and S. W. Lee, “A chromaticity space for specularly, illumination color- and illumination pose-invariant 3-d object recognition,” in *IEEE International Conference on Computer Vision (ICCV)*, 1998.
- [58] G. Healey and D. Slater, “Global color constancy: recognition of objects by use of illumination-invariant properties of color distributions,” *Journal of the Optical Society of America A*, 1994.
- [59] T. Gevers and A. W. M. Smeulders, “Color-based object recognition,” *Pattern Recognition*, 1999.

REFERENCES

- [60] G. West and M. H. Brill, “Necessary and sufficient conditions for von kries chromatic adaptation to give color constancy,” *Journal of Mathematical Biology*, 1982.
- [61] M. D’Zmura and P. Lennie, “Mechanisms of color constancy,” *Journal of the Optical Society of America A*, 1986.
- [62] G. D. Finlayson, M. S. Drew, and B. V. Funt, “Spectral sharpening: sensor transformations for improved color constancy,” *Journal of the Optical Society of America A*, 1994.
- [63] Q. Zaidi, “Identification of illuminant and object colors: heuristic-based algorithms,” *Journal of the Optical Society of America A*, 1998.
- [64] B. V. Funt and B. C. Lewis, “Diagonal versus affine transformations for color correction,” *Journal of the Optical Society of America A*, 2000.
- [65] G. D. Finlayson, M. S. Drew, and B. V. Funt, “Color constancy: generalized diagonal transforms suffice,” *Journal of the Optical Society of America A*, 1994.
- [66] E. Miller and K. Tieu, “Color eigenflows: statistical modeling of joint color changes,” in *IEEE International Conference on Computer Vision (ICCV)*, 2001.
- [67] M. Swain and D. Ballard, “Indexing via color histograms,” in *IEEE International Conference on Computer Vision (ICCV)*, 1990.
- [68] B. Funt and G. Finlayson, “Color constant color indexing,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 1995.
- [69] T. Gevers, “Robust segmentation and tracking of colored objects in video,” *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 2004.
- [70] T. Gevers and A. W. M. Smeulders, “Pictoseek: combining color and shape invariant features for image retrieval,” *IEEE Transactions on Image Processing (TIP)*, 2000.
- [71] J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus, “Learning color names for real-world applications,” *IEEE Transactions on Image Processing (TIP)*, 2009.

- [72] F. Porikli, “Inter-camera color calibration by correlation model function,” in *Proceedings. International Conference on Image Processing (ICIP)*, 2003.
- [73] O. Javed, K. Shafique, and M. Shah, “Appearance modeling for tracking in multiple non-overlapping cameras,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [74] B. Prosser, S. Gong, and T. Xiang, “Multi-camera matching using bi-directional cumulative brightness transfer functions,” in *Proceedings of the British Machine Vision Conference (BMVC)*, 2008.
- [75] A. Datta, L. Brown, R. Feris, and S. Pankanti, “Appearance modeling for person re-identification using weighted brightness transfer functions,” in *International Conference on Pattern Recognition (ICPR)*, 2012.
- [76] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, “Local fisher discriminant analysis for pedestrian re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [77] W.-S. Zheng, S. Gong, and T. Xiang, “Re-identification by relative distance comparison,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2013.
- [78] M. Dikmen, E. Akbas, T. S. Huang, and N. Ahuja, “Pedestrian recognition with a learned metric,” in *Proceedings of Asian Conference on Computer Vision (ACCV)*, 2010.
- [79] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” *CoRR*, vol. abs/1512.00567, 2015.
- [80] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *CoRR*, vol. abs/1512.03385, 2015.
- [81] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *CoRR*, vol. abs/1409.1556, 2014.

REFERENCES

- [82] R. K. Srivastava, K. Greff, and J. Schmidhuber, “Training very deep networks,” in *Advances in Neural Information Processing Systems (NIPS)*, 2015.
- [83] G. E. Hinton, “Connectionist learning procedures,” *Artificial intelligence*, 1989.
- [84] J. L. McClelland, D. E. Rumelhart, P. R. Group, *et al.*, “Parallel distributed processing,” *Explorations in the microstructure of cognition*, 1986.
- [85] Y. Bengio, “Learning deep architectures for AI,” *Foundations and trends in Machine Learning*, 2009.
- [86] T. Serre, G. Kreiman, M. Kouh, C. Cadieu, U. Knoblich, and T. Poggio, “A quantitative theory of immediate visual recognition,” *Progress in brain research*, 2007.
- [87] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, 1994.
- [88] G. Hinton, S. Osindero, and Y. W. Teh, “A fast learning algorithm for deep belief nets,” *Neural computation*, 2006.
- [89] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, 2006.
- [90] H. Lee, C. Ekanadham, and A. Y. Ng, “Sparse deep belief net model for visual area v2,” in *Advances in neural information processing systems (NIPS)*, 2008.
- [91] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, “Greedy layer-wise training of deep networks,” *Advances in neural information processing systems (NIPS)*, 2007.
- [92] Y. LeCun, L. Bottou, G. Orr, and K. Muller, “Efficient backprop,” in *Neural Networks: Tricks of the trade*, 1998.
- [93] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” *CoRR*, vol. abs/1502.01852, 2015.

REFERENCES

- [94] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *CoRR*, vol. abs/1502.03167, 2015.
- [95] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the International conference on Machine learning (ICML)*, 2008.
- [96] Z. Zuo and G. Wang, “Learning discriminative hierarchical features for object recognition,” *IEEE Signal Processing Letters*, 2014.
- [97] J. Yang, K. Yu, Y. Gong, and T. Huang, “Linear spatial pyramid matching using sparse coding for image classification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [98] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. A. Manzagol, “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,” *The Journal of Machine Learning Research (JMLR)*, 2010.
- [99] M. Chen, K. Q. Weinberger, F. Sha, and Y. Bengio, “Marginalized denoising autoencoders for nonlinear representations,” in *Proceedings of the International conference on Machine learning (ICML)*, 2014.
- [100] M. Chen, Z. Xu, K. Q. Weinberger, and F. Sha, “Marginalized denoising autoencoders for domain adaptation,” in *Proceedings of the International conference on Machine learning (ICML)*, 2012.
- [101] L. Fei-Fei and P. Perona, “A bayesian hierarchical model for learning natural scene categories,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [102] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

REFERENCES

- [103] K. Yu, T. Zhang, and Y. Gong, “Nonlinear learning using local coordinate coding,” in *Advances in neural information processing systems (NIPS)*, 2009.
- [104] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, “Locality-constrained linear coding for image classification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.
- [105] T. Liu, G. Wang, L. Wang, and K. L. Chan, “Visual tracking via temporally smooth sparse coding,” *IEEE Signal Processing Letters*, 2015.
- [106] A. Shahroudy, T. T. Ng, Q. Yang, and G. Wang, “Multimodal multipart learning for action recognition in depth videos,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2015.
- [107] E. Xing, R. Yan, and A. G. Hauptmann, “Mining associated text and images with dual-wing harmoniums,” in *Proceedings of the 21th Annual Conf. on Uncertainty in Artificial Intelligence (UAI)*, 2005.
- [108] G. E. Hinton and R. R. Salakhutdinov, “Replicated softmax: an undirected topic model,” in *Advances in Neural Information Processing Systems (NIPS)*, 2009.
- [109] R. Salakhutdinov and G. Hinton, “Learning a nonlinear embedding by preserving class neighbourhood structure,” in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2007.
- [110] H. Larochelle and Y. Bengio, “Classification using discriminative restricted Boltzmann machines,” in *International Conference on Machine Learning (ICML)*, 2008.
- [111] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems (NIPS)*, 2012.
- [112] B. Shuai, G. Wang, Z. Zuo, B. Wang, and L. Zhao, “Integrating parametric and non-parametric models for scene labeling,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

REFERENCES

- [113] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [114] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, “Translating videos to natural language using deep recurrent neural networks,” *arXiv preprint arXiv:1412.4729*, 2014.
- [115] X. Chen and C. L. Zitnick, “Learning a recurrent visual representation for image caption generation,” *arXiv preprint arXiv:1411.5654*, 2014.
- [116] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, and R. K. Ward, “Deep sentence embedding using the long short term memory network: Analysis and application to information retrieval,” *CoRR*, vol. abs/1502.06922, 2015.
- [117] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” *CoRR*, vol. abs/1409.3215, 2014.
- [118] A. Graves and N. Jaitly, “Towards end-to-end speech recognition with recurrent neural networks,” in *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.
- [119] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent, “Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription,” in *International Conference on Machine Learning (ICML)*, 2012.
- [120] W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki, “Scene labeling with lstm recurrent neural networks,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [121] P. H. O. Pinheiro and R. Collobert, “Recurrent convolutional neural networks for scene labeling,” in *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.

REFERENCES

- [122] A. Shahroudy, J. Liu, T. T. Ng, and G. Wang, “Ntu rgb+d: A large scale dataset for 3d human activity analysis,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [123] J. Liu, A. Shahroudy, D. Xu, and G. Wang, “Spatio-temporal lstm with trust gates for 3d human action recognition,” in *European Conference on Computer Vision (ECCV)*, 2016.
- [124] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, 1997.
- [125] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, “Long-term recurrent convolutional networks for visual recognition and description,” in *CoRR*, vol. abs/1403.2124, 2014.
- [126] A. Karpathy, J. Johnson, and F. Li, “Visualizing and understanding recurrent networks,” *CoRR*, vol. abs/1506.02078, 2015.
- [127] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A search space odyssey,” *CoRR*, vol. abs/1503.04069, 2015.
- [128] R. Jzefowicz, W. Zaremba, and I. Sutskever, “An empirical exploration of recurrent network architectures,” in *Proceedings of the International conference on Machine learning, (ICML)*, 2015.
- [129] S. Fernandez, A. Graves, and J. Schmidhuber, “An application of recurrent neural networks to discriminative keyword spotting,” in *International Conference on Artificial Neural Networks ICANN 2007*, 2007.
- [130] M. Stollenga, W. Byeon, M. Liwicki, and J. Schmidhuber, “Parallel multi-dimensional lstm, with application to fast biomedical volumetric image segmentation,” *CoRR*, vol. abs/1506.07452, 2015.
- [131] P. Werbos, “Backpropagation through time: what does it do and how to do it,” in *Proceedings of IEEE*, 1990.

REFERENCES

- [132] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, “Deep captioning with multimodal recurrent neural networks (m-rnn),” *arXiv preprint arXiv:1412.6632*, 2014.
- [133] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, “Beyond short snippets: Deep networks for video classification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [134] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra, “Draw: A recurrent neural network for image generation,” *arXiv preprint arXiv:1502.04623*, 2015.
- [135] V. Mnih, N. Heess, A. Graves, *et al.*, “Recurrent models of visual attention,” in *Advances in Neural Information Processing Systems (NIPS)*, 2014.
- [136] D. Eigen, J. Rolfe, R. Fergus, and Y. LeCun, “Understanding deep architectures using a recursive convolutional network,” *arXiv preprint arXiv:1312.1847*, 2013.
- [137] R. Zhao, W. Ouyang, and X. Wang, “Learning mid-level filters for person re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [138] E. Ahmed, M. Jones, and T. K. Marks, “An improved deep learning architecture for person re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [139] 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)*, 2016.
- [140] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang, “Joint learning of single-image and cross-image representations for person re-identification,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

REFERENCES

- [141] T. Xiao, H. Li, W. Ouyang, and X. Wang, “Learning deep feature representations with domain guided dropout for person re-identification,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [142] D. Yi, Z. Lei, S. Liao, and S. Z. Li, “Deep metric learning for person re-identification,” *Proceedings of International Conference on Pattern Recognition (ICPR)*, 2014.
- [143] R. Hadsell, S. Chopra, and Y. LeCun, “Dimensionality reduction by learning an invariant mapping,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR) 2006*, 2006.
- [144] F. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: continual prediction with lstm,” in *International Conference on Artificial Neural Networks, (ICANN) 1999*, 1999.
- [145] H. Wang, S. Gong, and T. Xiang, “Unsupervised learning of generative topic saliency for person re-identification,” in *Proceedings of the British Machine Vision Conference (BMVC)*, 2014.
- [146] A. Gijsenij and T. Gevers, “Color constancy using natural image statistics and scene semantics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2011.
- [147] M. Drew, J. Wei, and Z. Li, “Illumination-invariant color object recognition via compressed chromaticity histograms of color-channel-normalized images,” in *International Conference on Computer Vision (ICCV)*, 1998.
- [148] G. D. Finlayson, M. S. Drew, and B. V. Funt, “Color constancy: enhancing von kries adaption via sensor transformations,” in *IS&T/SPIE’s Symposium on Electronic Imaging: Science and Technology*, 1993.
- [149] B. A. Olshausen and D. J. Field, “Sparse coding of sensory inputs,” *Current opinion in neurobiology*, 2004.

REFERENCES

- [150] M. A. Ranzato, Y. Boureau, and Y. LeCun, “Sparse feature learning for deep belief networks,” in *Advances in Neural Information Processing Systems (NIPS)*, 2008.
- [151] A. Wang, J. Lu, G. Wang, J. Cai, and T. J. Cham, “Multi-modal unsupervised feature learning for rgb-d scene labeling,” in *European Conference on Computer Vision (ECCV)*, 2014.
- [152] H. Lee, A. Battle, R. Raina, and A. Y. Ng, “Efficient sparse coding algorithms,” in *Advances in neural information processing systems (NIPS)*, 2006.
- [153] S. I. Lee, H. Lee, P. Abbeel, and A. Y. Ng, “Efficient L_1 regularized logistic regression,” in *Proceedings of the National Conference on Artificial Intelligence*, 2006.
- [154] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online dictionary learning for sparse coding,” in *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, pp. 689–696, ACM, 2009.
- [155] A. Coates, A. Y. Ng, and H. Lee, “An analysis of single-layer networks in unsupervised feature learning,” in *International Conference on Artificial Intelligence and Statistics (ICAIS)*, 2011.
- [156] A. Gijsenij, R. Lu, and T. Gevers, “Color constancy for multiple light sources,” *IEEE Transactions on Image Processing (TIP)*, 2012.
- [157] K. E. A. Van de Sande, T. Gevers, and C. G. M. Snoek, “Evaluating color descriptors for object and scene recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2010.
- [158] Y. Jia, “Caffe: An open source convolutional architecture for fast feature embedding.” <http://caffe.berkeleyvision.org/>, 2013.
- [159] M. Hirzer, P. M. Roth, M. Köstinger, and H. Bischof, “Relaxed pairwise learned metric for person re-identification,” in *European Conference on Computer Vision (ECCV)*, 2012.

- [160] L. Zheng, S. Wang, L. Tian, F. He, Z. Liu, and Q. Tian, “Query-adaptive late fusion for image search and person re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [161] Y. Shen, W. Lin, J. Yan, M. Xu, J. Wu, and J. Wang, “Person re-identification with correspondence structure learning,” in *The IEEE International Conference on Computer Vision (ICCV)*, 2015.
- [162] C. Su, F. Yang, S. Zhang, Q. Tian, L. S. Davis, and W. Gao, “Multi-task learning with low rank attribute embedding for person re-identification,” in *The IEEE International Conference on Computer Vision (ICCV)*, 2015.
- [163] Z. Shi, T. M. Hospedales, and T. Xiang, “Transferring a semantic representation for person re-identification and search,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [164] 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)*, 2016.
- [165] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato, “Hierarchical gaussian descriptor for person re-identification,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [166] 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)*, 2016.
- [167] H. Wang, S. Gong, and T. Xiang, “Highly efficient regression for scalable person re-identification,” in *Proceedings of the British Machine Vision Conference (BMVC)*, 2016.
- [168] 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)*, 2016.

REFERENCES

- [169] L. Bazzani, M. Cristani, A. Perina, and V. Murino, “Multiple-shot person re-identification by chromatic and epitomic analyses,” *Pattern Recognition Letters*, 2012.
- [170] X. Liu, M. Song, D. Tao, X. Zhou, C. Chen, and J. Bu, “Semi-supervised coupled dictionary learning for person re-identification,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [171] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN features off-the-shelf: an astounding baseline for recognition,” *CoRR*, 2014.
- [172] R. R. Varior, G. Wang, J. Lu, and T. Liu, “Learning invariant color features for person re-identification,” *IEEE Transactions on Image Processing (TIP)*, 2016.
- [173] S. Z. Chen, C. C. Guo, and J. H. Lai, “Deep ranking for person re-identification via joint representation learning,” *IEEE Transactions on Image Processing (T-IP)*, 2016.
- [174] S. Ding, L. Lin, G. Wang, and H. Chao, “Deep feature learning with relative distance comparison for person re-identification,” *Pattern Recognition*, 2015.
- [175] F. Gers and J. Schmidhuber, “Recurrent nets that time and count,” in *Proceedings of the International Joint Conference on Neural Networks, (IJCNN) 2000*, 2000.
- [176] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM and other neural network architectures,” *Neural Networks*, 2005.
- [177] Y. Bengio, “Practical recommendations for gradient-based training of deep architectures,” in *Neural Networks: Tricks of the Trade*, Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012.
- [178] Y. N. Dauphin, H. de Vries, J. Chung, and Y. Bengio, “Rmsprop and equilibrated adaptive learning rates for non-convex optimization,” *CoRR*, vol. abs/1502.04390, 2015.

REFERENCES

- [179] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, “Learning to rank in person re-identification with metric ensembles,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [180] M. Guillaumin, J. Verbeek, and C. Schmid, “Is that you? metric learning approaches for face identification,” in *IEEE 12th International Conference on Computer Vision, (ICCV)*, 2009.
- [181] 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 (ECCV)*, 2016.
- [182] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, “High-performance neural networks for visual object classification,” *CoRR*, vol. abs/1102.0183, 2011.
- [183] D. C. Ciresan, U. Meier, and J. Schmidhuber, “Multi-column deep neural networks for image classification,” *CoRR*, vol. abs/1202.2745, 2012.
- [184] A. Vedaldi and K. Lenc, “Matconvnet – convolutional neural networks for matlab,” 2015.
- [185] R. R. Varior, M. Haloi, and G. Wang, “Gated siamese convolutional neural network architecture for human re-identification,” in *European Conference on Computer Vision (ECCV)*, 2016.
- [186] C. Wengert, M. Douze, and H. Jégou, “Bag-of-colors for improved image search,” in *MM 2011 - 19th ACM International Conference on Multimedia*, pp. 1437–1440, ACM, 2011.
- [187] H. Jégou, M. Douze, and C. Schmid, “Hamming embedding and weak geometric consistency for large scale image search,” in *European Conference on Computer Vision (ECCV)*, 2008.
- [188] D. Nistér and H. Stewénus, “Scalable recognition with a vocabulary tree,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

REFERENCES

- [189] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, “Neural codes for image retrieval,” in *European Conference on Computer Vision (ECCV)*, 2014.
- [190] H. Jégou, M. Douze, and C. Schmid, “Improving bag-of-features for large scale image search,” *International Journal of Computer Vision (IJCV)*, 2010.
- [191] A. Oliva and A. Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *International journal of computer vision (IJCV)*, 2001.