Summary

Visual search and recognition underpins numerous applications including management of multimedia content, mobile commerce, surveillance, navigation, robotics and many others. However the task is still challenging predominantly due to the variability of object appearance and ever increasing size of the databases, often exceeding billions of images. The objective of this thesis is to develop a robust, compact and discriminative image representation suitable for tasks of visual search.

This thesis contributes to four research areas. First we propose a novel method, named Robust Visual Descriptor (RVD), for deriving a compact and robust representation of image content which significantly advances state of the art and delivers world-class performance. In our approach, the local descriptors are assigned to multiple cluster centres with rank weights leading to a stable and reliable global image representation. Residual vectors are then computed in each cluster, normalized using a direction preserving normalization and aggregated based on the neighbourhood rank information.

We then propose two extensions to the core RVD descriptor. The first one consists of decorrelating weighted residual vectors by applying cluster level PCA before aggregation. In the second extension, the weighted residual vectors are whitened in each cluster before aggregation, leading to a balanced energy distribution in each dimension and improved performance.

Compressing floating point global signatures to binary codes improves storage requirements and matching speed for large scale image retrieval tasks. Our third contribution is to derive a compact and robust binary image signature from the core RVD representation. In addition, we propose a novel binary descriptors matching algorithm, PCAE with Weighted Hamming distance (PCAE+WH), to minimize the quantization loss associated with converting floating point vector to discrete binary codes.

In the context of industry work on Compact descriptors for Visual Search (CDVS) and its standardization in MPEG (ISO), we propose a scalable RVD representation. The bitrate scalability is achieved by employing novel Cluster Selection and Bit Selection mechanisms which support interoperable binary RVD representations. Moreover, we propose a very efficient and effective score function based on weighted Hamming distance, to compute similarity between two binary representations.

Our fourth contribution is to develop an image classification system based on RVD representation. We introduce an effective method to incorporate second order statistics in the original RVD framework.

Key words: Image retrieval, aggregation of local descriptors, global descriptors, Fisher Vectors, Vector of Locally Aggregated descriptors, Robust visual descriptor (RVD), Robust visual descriptor with Whitening (RVD-W),
Email: s.husain@surrey.ac.uk
WWW: http://www.eps.surrey.ac.uk/
Acknowledgements

I would like to thank my supervisors Professor Miroslaw Bober and Professor Josef Kittler for their guidance and continuous support of my PhD study and research. Their ocean of knowledge, as well as enthusiasm, passion and devotion have played a pivotal role in shaping the direction of my research.

I would like to thank Professor John Illingworth, Professor Richard Bowden and Professor Mark Plumbley for giving useful advice and feedback about my research. Also I thank Dr Stavros and Dr Karol for helping me to develop Mobile Visual Search system for MPEG CDVS proposal.

Finally, I would like to thank my parents Shaista and Wilayat Husain for supporting me throughout my life. They have been standing by me all the time, believing in me and my ability to achieve more and more in the field of higher education.
# Contents

## Nomenclature

1 Introduction 1

1.1 Objectives and Motivation ........................................ 1

1.2 Challenges ................................................................ 3

1.3 Research contributions ............................................. 6

1.4 Thesis outline .......................................................... 7

1.5 Publication .............................................................. 9

2 Literature Review ....................................................... 11

2.1 Keypoints detection .................................................. 12

2.2 Local descriptors extraction ....................................... 14

2.3 Global image representation ....................................... 16

2.3.1 Bag of Words ....................................................... 16

2.3.2 Fisher Vectors ....................................................... 18

2.3.3 VLAD and VLAT .................................................. 20

2.3.4 Triangulation embedding (Temb) ............................ 23

2.3.5 Global descriptors based on Convolutional Neural Networks (CNN) .................................................. 25

2.4 Compression of global descriptors ......................... 26

2.5 Comparison of global descriptors ....................... 30

3 Robust Visual Descriptor ........................................... 33

3.1 Rank-based multiple assignment ............................. 35

3.1.1 Direction preserving mapping function .............. 41

3.2 Intra stage processing of cluster level representation 44
### Contents

3.2.1 RVD with local PCA (RVD-P) ................................. 44
3.2.2 RVD with local Whitening (RVD-W) ......................... 46
3.2.3 PCA transformation and L1+Power normalization .......... 48

3.3 Compact global descriptor ...................................... 51

3.4 Experimental Setup and Evaluation ............................. 52

3.4.1 Datasets ..................................................... 53
3.4.2 Local descriptor extraction .................................. 54
3.4.3 Dimensionality reduction on local descriptors ............... 54
3.4.4 Vocabulary size ............................................. 55
3.4.5 Comparison of RVD/RVD-P/RVD-W/FV/Temb ................. 55
3.4.6 Optimized Product Quantization .............................. 57
3.4.7 Large scale experiments ................................. 57

3.5 Comparison with the state of the art ........................... 61

3.6 Summary ....................................................... 62

4 Binary RVD ................................................................ 65

4.1 Introduction ....................................................... 65

4.2 Compact Global Descriptor ........................................ 67

4.2.1 PCA Embedding Binarization (PCAE) ..................... 67
4.2.2 PCA Embedding with Random Rotations (PCAE+RR) .... 69
4.2.3 PCA Embedding with Iterative Quantization (PCAE+ITQ) 69
4.2.4 PCAE with Weighted Hamming distance (PCAE+WH) .... 71
4.2.5 Computation of Weights Φ and Ψ ......................... 72

4.3 Experimental Setup and Evaluation ............................. 75

4.3.1 Datasets ..................................................... 75
4.3.2 Local descriptor extraction .................................. 75
4.3.3 Comparison of RVD/RVD-P/RVD-W/FV/VLAD .......... 75
4.3.4 PCAE with Weighted Hamming distance .................... 77

4.4 Binary RVD versus the state of the art .......................... 80

4.5 Conclusion ....................................................... 81
## 5 Mobile Visual Search based on Scalable RVD

5.1 Modern Mobile Visual Search System ........................................... 85
5.2 Compact Descriptors for Visual Search (CDVS) standard ............... 86
  5.2.1 Compact descriptor extraction ............................................... 86
  5.2.2 CDVS evaluation framework [4] .......................................... 87
  5.2.3 Retrieval architecture ....................................................... 87
  5.2.4 Pairwise matching architecture ........................................... 89
5.3 Scalable Robust Visual descriptor (RVD) .................................... 89
  5.3.1 Component Cluster selection .............................................. 91
  5.3.2 Binarizing RVD ............................................................. 92
  5.3.3 Bit Selection in RVD ...................................................... 92
  5.3.4 RVD matching ............................................................. 96
5.4 Experimental Setup and Evaluation ........................................... 97
  5.4.1 Datasets ........................................................................... 97
  5.4.2 Local descriptor extraction ................................................. 98
  5.4.3 Memory footprint of RVD, SCFV and REVV .......................... 99
  5.4.4 Scalable RVD ............................................................... 100
  5.4.5 Bit Selection methods ...................................................... 101
5.5 Comparison with the state of the art ......................................... 103
  5.5.1 Comparison in global descriptor matching ............................ 103
  5.5.2 Comparison in global and local descriptor matching .............. 106
  5.5.3 Overall comparison of Mobile Visual Search systems performance 106
5.6 Conclusion ........................................................................... 109

## 6 Image Classification based on higher order RVD

6.1 Global Descriptors for classification ......................................... 113
  6.1.1 Extending VLAD to H-VLAD ............................................... 113
  6.1.2 Fisher Vectors .................................................................... 114
  6.1.3 Extending the RVD for classification tasks ............................ 115
  6.1.4 Improving RVD-W representation for object classification ...... 116
6.2 Experiments ............................................................................ 118
  6.2.1 Evaluation on PASCAL VOC 2007 ........................................... 119
List of Figures

1.1 Visual Search Applications .................................................. 4
1.2 Robustness challenges .......................................................... 5
1.3 Scalability challenges .............................................................. 5

2.1 Retrieval by matching global descriptors. ................................. 12
2.2 Bag of Words model. ................................................................. 17
2.3 Fisher Vectors ................................................................. 19
2.4 Vector of Locally Aggregated Descriptors .............................. 20
2.5 Triangulation Embedding ...................................................... 24

3.1 RVD-W extraction pipeline .................................................. 34
3.2 Impact of rank-based weighting on performance ...................... 37
3.3 Fisher Vectors and RVD Statistics ........................................ 38
3.4 Impact of rank-based aggregation on performance ................. 40
3.5 RVD Statistics ...................................................................... 41
3.6 Advantage of normalizing the residual vectors before aggregation 43
3.7 Impact of L1-normalization of residual vectors on performance 44
3.8 RVD aggregation ................................................................. 45
3.9 Comparison of cluster level PCA and LDA ......................... 47
3.10 Energy distribution in each dimension of residual vectors ...... 48
3.11 Global descriptors comparison .............................................. 48
3.12 Histogram of Euclidean similarity between matching and non-matching descriptors, for three post-PCA normalization methods (a) Whitening (b) P-L2 (c) L1-P. .............................................. 49
3.13 (a) Impact of power-norm exponent $\beta$ on Holidays dataset using $D' = 8192$ descriptor, (b) Impact of $\beta$ on Holidays dataset using $D' = 128$ descriptor. Comparison of post PCA normalization methods on (c) Holidays and (d) Holidays1M .............................................. 51
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.14</td>
<td>SIFT dimensionality reduction and Codebook size</td>
<td>54</td>
</tr>
<tr>
<td>3.15</td>
<td>Global descriptors comparison</td>
<td>56</td>
</tr>
<tr>
<td>3.16</td>
<td>Large scale experiments</td>
<td>59</td>
</tr>
<tr>
<td>3.17</td>
<td>Example retrieval results for the RVD-W and FV descriptors on the Oxford1M dataset</td>
<td>60</td>
</tr>
<tr>
<td>4.1</td>
<td>PCA Embedding Binarisation pipeline</td>
<td>68</td>
</tr>
<tr>
<td>4.2</td>
<td>Energy of different dimensions of RVD signatures for PCA (a), RR (b) and ITQ (c)</td>
<td>69</td>
</tr>
<tr>
<td>4.3</td>
<td>Binary Embedding Methods</td>
<td>71</td>
</tr>
<tr>
<td>4.4</td>
<td>Distribution of the projected values of 1 million images</td>
<td>72</td>
</tr>
<tr>
<td>4.5</td>
<td>PCAE with Weighted Hamming distance pipeline</td>
<td>74</td>
</tr>
<tr>
<td>4.6</td>
<td>Binary global descriptors comparison</td>
<td>76</td>
</tr>
<tr>
<td>4.7</td>
<td>Comparison of PCAE+H and PCAE+WH methods</td>
<td>78</td>
</tr>
<tr>
<td>4.8</td>
<td>Binarization and matching of global descriptors using PCAE+WH</td>
<td>79</td>
</tr>
<tr>
<td>5.1</td>
<td>Mobile Visual Search System</td>
<td>85</td>
</tr>
<tr>
<td>5.2</td>
<td>Compact descriptor extraction pipeline</td>
<td>86</td>
</tr>
<tr>
<td>5.3</td>
<td>CDVS retrieval pipeline</td>
<td>88</td>
</tr>
<tr>
<td>5.4</td>
<td>CDVS pairwise matching pipeline</td>
<td>88</td>
</tr>
<tr>
<td>5.5</td>
<td>Extraction and matching of scalable Robust Visual Descriptor</td>
<td>90</td>
</tr>
<tr>
<td>5.6</td>
<td>Illustration of reliability factor $CS_j$ values in an example image</td>
<td>91</td>
</tr>
<tr>
<td>5.7</td>
<td>Cluster Selection RVD</td>
<td>93</td>
</tr>
<tr>
<td>5.8</td>
<td>Distribution of $Pr(h</td>
<td>m)$ and $Pr(h</td>
</tr>
<tr>
<td>5.9</td>
<td>Example of images in MPEG dataset</td>
<td>98</td>
</tr>
<tr>
<td>5.10</td>
<td>Bit Selection methods</td>
<td>102</td>
</tr>
<tr>
<td>5.11</td>
<td>Pairwise matching using global descriptor</td>
<td>104</td>
</tr>
<tr>
<td>5.12</td>
<td>Image retrieval using global descriptor</td>
<td>105</td>
</tr>
<tr>
<td>5.13</td>
<td>Pairwise matching results of MVS systems</td>
<td>107</td>
</tr>
<tr>
<td>5.14</td>
<td>Retrieval performance of MVS systems</td>
<td>108</td>
</tr>
<tr>
<td>5.15</td>
<td>Performance of MVS systems within the CDVS framework</td>
<td>109</td>
</tr>
</tbody>
</table>
6.1 Image classification pipeline ................................................. 112
6.2 Power normalization of RVD-W representation ..................... 117
6.3 Impact of Power normalization factor alpha on the classification accuracy ................................................................. 117
6.4 SIFT dimensionality reduction and Codebook size .................. 120
6.5 Global descriptor comparison on PASCAL VOC dataset .......... 121
6.6 Impact of Spatial Pyramid Matching on PASCAL VOC dataset .... 122
6.7 Spatial Pyramid Matching Example ....................................... 123
6.8 Image classification performance on PASCAL VOC dataset ...... 124
6.9 Global descriptor comparison on Caltech256 dataset ............... 125
6.10 Impact of Spatial Pyramid Matching on Caltech256 dataset ...... 126
6.11 Global descriptor comparison on MIT Scene 67 dataset .......... 126
6.12 Impact of Spatial Pyramid Matching on MIT Scene 67 dataset ... 127
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Comparison of global image representations</td>
<td>31</td>
</tr>
<tr>
<td>3.1</td>
<td>Dimensionality reduction of RVD-W vectors</td>
<td>55</td>
</tr>
<tr>
<td>3.2</td>
<td>RVD, RVD-P, RVD-W and FV performance using 16 bytes codes</td>
<td>58</td>
</tr>
<tr>
<td>3.3</td>
<td>Comparison of RVD-W with the state-of-the-art using full dimensional vectors</td>
<td>63</td>
</tr>
<tr>
<td>3.4</td>
<td>Comparison of RVD-W with the state-of-the-art using 96/128 dimensional vectors</td>
<td>64</td>
</tr>
<tr>
<td>3.5</td>
<td>Comparison of RVD-W with the state-of-the-art with compact codes via Product Quantization (PQ)</td>
<td>64</td>
</tr>
<tr>
<td>4.1</td>
<td>Performance of binary global representations on state-of-the-art datasets</td>
<td>80</td>
</tr>
<tr>
<td>5.1</td>
<td>Mean and median of KI-divergence scores</td>
<td>96</td>
</tr>
<tr>
<td>5.2</td>
<td>Number of matching and non-matching image pairs in MPEG CDVS dataset for pairwise matching experiments</td>
<td>99</td>
</tr>
<tr>
<td>5.3</td>
<td>Number of query images and database images in MPEG CDVS dataset for retrieval experiments</td>
<td>99</td>
</tr>
<tr>
<td>5.4</td>
<td>Memory footprint of RVD, SCFV and REVV</td>
<td>100</td>
</tr>
<tr>
<td>5.5</td>
<td>The bitrates of the proposed RVD</td>
<td>101</td>
</tr>
<tr>
<td>6.1</td>
<td>Performance of VLAD, VLAT and H-VLAD</td>
<td>114</td>
</tr>
<tr>
<td>6.2</td>
<td>Spatial Pyramid matching</td>
<td>118</td>
</tr>
<tr>
<td>6.3</td>
<td>Comparison of SIFT and RootSIFT descriptors</td>
<td>119</td>
</tr>
<tr>
<td>6.4</td>
<td>Performance of image representations on state-of-the-art datasets, VQ: Vector Quantization, LLC: Linear Local Coding, SV: Supervectors, FV: Fisher Vectors and RVD-W</td>
<td>128</td>
</tr>
</tbody>
</table>
Nomenclature

List of Symbols

\( \phi_{\Delta t} \)  
Triangulation Embedding representation

\( \varrho_j \)  
Skewness of \( j \)-th cluster

\( \zeta_\Theta \)  
Fisher vector representation

\( \eta_j \)  
Mean of weighted residual vectors \( r_{tj} \) belonging to cluster \( j \)

\( \gamma \)  
Nearest cluster rank

\( \kappa \)  
Level of the vocabulary tree

\( \mu_j \)  
Cluster centre \( j \)

\( \omega_j \)  
Weight of cluster \( j \)

\( \psi_d \)  
Democratic aggregation

\( \psi_s \)  
Sum aggregation

\( \Sigma_j \)  
Covariance matrix of cluster \( j \)

\( \tau_{tj} \)  
Assignment weight of descriptor \( x_t \) to cluster \( j \)

\( \varphi_j \)  
Mean of residual vectors belonging to cluster \( j \)

\( \zeta_j \)  
Cluster level representation of global descriptor

\( CS_j \)  
Reliability factor of cluster \( j \)

\( CS_{th} \)  
Cluster selection threshold

\( D \)  
Global descriptor dimensionality

\( d \)  
Dimensionality of local descriptor

\( D' \)  
Dimensionality of global descriptor after PCA transformation

\( d' \)  
Local descriptor dimensionality after dimensionality reduction

\( I \)  
Total number of images in training dataset
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K )</td>
<td>Maximum number of clusters a descriptor is assigned to</td>
</tr>
<tr>
<td>( M )</td>
<td>Median threshold for Hamming Embedding</td>
</tr>
<tr>
<td>( n )</td>
<td>Visual vocabulary size</td>
</tr>
<tr>
<td>( N_j )</td>
<td>Number of local descriptors in cluster ( j )</td>
</tr>
<tr>
<td>( NN^K_\gamma(x_t) )</td>
<td>Returns the cluster index that is rank ( \gamma ) from ( x_t )</td>
</tr>
<tr>
<td>( P )</td>
<td>PCA matrix</td>
</tr>
<tr>
<td>( P_{lda,j} )</td>
<td>LDA transformation matrix for cluster ( j )</td>
</tr>
<tr>
<td>( Q )</td>
<td>Random orthogonal matrix</td>
</tr>
<tr>
<td>( R )</td>
<td>RVD global descriptor</td>
</tr>
<tr>
<td>( R^p )</td>
<td>PCA projected RVD descriptor</td>
</tr>
<tr>
<td>( R^p )</td>
<td>RVD-P global descriptor</td>
</tr>
<tr>
<td>( R^{w\rho} )</td>
<td>PCA transformed RVD-W descriptor</td>
</tr>
<tr>
<td>( R^{wb} )</td>
<td>Binary RVD-W descriptor</td>
</tr>
<tr>
<td>( R^{whit} )</td>
<td>Whitened RVD-W descriptor</td>
</tr>
<tr>
<td>( R^{wl} )</td>
<td>L1-P transformed RVD-W descriptor</td>
</tr>
<tr>
<td>( R^{wo} )</td>
<td>L1-normalized RVD-W descriptor</td>
</tr>
<tr>
<td>( R^{wq} )</td>
<td>Randomly rotated RVD-W descriptor</td>
</tr>
<tr>
<td>( R^w )</td>
<td>RVD-W global descriptor</td>
</tr>
<tr>
<td>( R_0 )</td>
<td>Mean of the signatures of ( R^w )</td>
</tr>
<tr>
<td>( r_{tj} )</td>
<td>Weighted residual vector belonging to cluster ( j )</td>
</tr>
<tr>
<td>( s_{tj} )</td>
<td>Set of normalized residual vectors in Triangulation Embedding</td>
</tr>
<tr>
<td>( T )</td>
<td>Set of local descriptors extracted from an image</td>
</tr>
<tr>
<td>( T_I )</td>
<td>Set of local descriptors extracted from training images</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Objectives and Motivation

Research in visual search has become one of the most popular directions in the area of pattern analysis and machine intelligence. The explosive growth in the multimedia industry has made explicit the need for effective and computationally efficient content search systems. Early solutions, such as the text-based and semantic methods, are unable to cope with this huge amount of visual content, stimulating intensive research in the visual search domain.

The use of visual search in devices such as mobile phones, tablets and computers has expedited in the past decade, with several well-known companies adopting the technology and developing novel applications. Examples of these systems are Google Goggles [2], Kooaba [1], CamFind [3] and Layar [3].

The aim of this thesis is to develop a robust, discriminative and compact image representation to underpin tasks of visual search and classification. Our main focus is on efficient retrieval of particular objects such as books, dvds, paintings, buildings and printed documents. We also consider recognizing categories of objects and scenes, such as person, cat, car, bottle, chair, bedroom and living room.

Below we highlight some fields of application that exploit visual search and classification. (Figure 1.1).
1.1. Objectives and Motivation

**Image search engine**: With the exponential increase of the image-capturing devices, the need for web-scale search engines based on visual cues has become significantly important. The job of a web-scale search engine is to find a particular object such as a famous building or a painting, specified by the user, in a huge amount of visual data. Google Images is an example of such a system.

**Mobile Visual Search**: A visual search system that allows mobile users to snap a picture of a real world object such as a statue or a product and then use this picture to retrieve useful information about the object.

**Mobile commerce**: Nowadays shopping via electronic commerce has become a habit, making life more comfortable for consumers. By using their mobile devices to scan the QR-codes or bar codes on posters of the products, consumers can purchase the desired products easily without the need to type product names. A consumer can also use their mobile phones to recognize and compare prices of products, such as books, DVDs or clothes, and can then purchase them directly from the mobile device. Google Goggles is an example of visual search system which searches an image database with a picture taken by a mobile device. Currently, it supports the search for landmarks, bar-codes, books, contact info, artwork, wines and logos.

**Automatic tagging and annotation of images**: Personal photographs can be labeled and annotated automatically with places or objects, for efficient search and navigation through a large corpus of personal image collections. Furthermore, they can be classified and grouped into classes of interest, for example indoor or outdoor scenes.

**Security**: Nowadays video surveillance plays an important role in ensuring public safety: most cites have thousands of close-circuit cameras. Currently, the visual data from these cameras must be scrutinized by human operators. A fast retrieval application with object recognition capabilities is required to make the requested data easily accessible.

**Augmented Automotive Navigation**: With automotive infotainment systems, real time object recognition can be used to provide serviceable information to the driver and passengers. An image sensor mounted on a car constantly scans the environment that surrounds the car and the system automatically recognizes nearest vehicle, buildings
and landmarks. Retrieved information, such as proximity to the nearby vehicles, lane drifts or landmark names, is then provided to car occupants thus improving the journey experience.

**Robotic Vision:** Autonomous robots equipped with high speed cameras can recognize objects in their surrounding to localize themselves and to enable automated interactions.

## 1.2 Challenges

Despite formidable efforts, the performance of existing visual search systems is still lacking in terms of robustness, processing speed and detection rate. Image retrieval needs to be as fast and accurate as text retrieval has become in search engines like Google and Bing.

The task of performing robust, accurate and scalable visual search is challenging on a number of fronts; here we list a few:

**Robustness:** An image retrieval system should be robust to the changes in the object’s appearance. Such changes can be caused by varying illumination conditions, digital artefacts (JPEG compression) and weather. The object can be scaled, rotated and imaged from different viewpoints. Furthermore, the system should handle partial occlusions caused by other objects located between the target object and the camera, as well as dissimilar backgrounds (clutter) surrounding objects of interest. Some examples of these are presented in Figure 1.2.

**Speed and scalability:** The rise of digital cameras and smart phones, the standardization of computers and multimedia, the ubiquity of data storage devices and the technological maturity of network infrastructure has exponentially increased the volumes of visual data available on-line (Figure 1.3). Social networking website Facebook declared the number of images uploaded to WWW as 250 billion in 2013, consuming approximately 6 petabytes of memory.\(^1\) The number of images uploaded to Flickr was

---

\(^1\) [http://www.theverge.com/2013/2/22/4016752/facebook-cold-storage-old-photos-prineville-data-center](http://www.theverge.com/2013/2/22/4016752/facebook-cold-storage-old-photos-prineville-data-center)
1.2. Challenges

(a) Image search engine  
(b) Mobile Visual Search  
(c) Product Search  
(d) Bar code scanning  
(e) Automatic tagging of images  
(f) Security  
(g) Augmented Automotive Navigation  
(g) Robotic Vision

Figure 1.1: Visual Search Applications
1.2. Challenges

- Viewpoint
- Illumination conditions
- Scale
- Partial Occlusion
- Rotation
- Cropping

Figure 1.2: Robustness challenges

Figure 1.3: Scalability challenges
1.3 Research contributions

This thesis contributes novel algorithms for robust and efficient visual search, recognition and classification suitable for web-scale databases.

The main contributions can be summarized as follows:

- **Core Robust Visual Descriptor (RVD) representation:** We develop a novel global descriptor for image retrieval and classification tasks. In our approach local descriptors are rank-assigned to multiple clusters. Residual vectors are then computed in each cluster, normalised using a direction-preserving normalization function and aggregated based on the neighbourhood rank. We also present a detailed experimental study to illustrate the effects of various new elements introduced into the RVD pipeline and to explain the factors contributing to its superior performance.

- **New intra stage processing of cluster level representations:** We further improve the performance of the core RVD representation by balancing the variance of residual vector directions, in order to maximize the discriminatory power of the global signature. This is achieved by introducing a novel intra stage pre-processing of the residual directions using cluster-wise PCA transform with a whitening operation. We call this representation RVD-W.

10 Billion in May 2015[^1] for Picasa it was 7 Billion, and for Photobucket 8 Billion (2011). According to Youtube statistics, 100 hours worth of videos are uploaded every minute. Searching content among billions of images in near-real time is an enormous task.

In addition to the above requirements, there are other constraints which depend on the type of system utilising visual search. For example a Mobile Visual Search system, which is executed on a mobile device, requires algorithms with low computational complexity to improve execution speeds and battery life and small memory footprint for low-cost hardware implementations.

• **Compact RVD-W signature**: To work with web-scale databases, the dimensionality of RVD-W descriptor is reduced via Principal Component Analysis. The compact global descriptors are indexed and searched effectively using Optimized Product Quantization and the Asymmetric Distance approach.

• **Binary global image representation**: We convert high dimensional global descriptors into compact binary signatures. Through detailed evaluation, we show that the RVD and RVD-W signatures outperform FV and VLAD in the binary domain. Furthermore, we propose a novel binary descriptors matching algorithm, PCAE with Weighted Hamming distance, to minimize the quantization loss of converting real valued vectors to discrete binary codes. Our binary signature delivers world-class performance.

• **Visual search system with scalable RVD**: In the context of industry work on Compact descriptors for Visual Search (CDVS), we have proposed a compact and scalable RVD signature with novel Cluster Selection and Bit Selection mechanisms. Our method improves significantly upon the MPEG CDVS Standard which is based on the Scalable Compressed Fisher Vector.

• **Image classification based on higher order RVD-W**: For image classification tasks we propose to extend the RVD-W framework with second order statistics (diagonal covariance) of the residual vectors. Furthermore, the RVD-W descriptor is incorporated within the Spatial Pyramid Matching (SPM) framework to capture more information regarding the structure of the scene in an image. Experimental results show that RVD-W outperforms state-of-the-art global representations on all image classification datasets.

### 1.4 Thesis outline

The structure of the thesis is as follows. Chapter 2 reviews the literature that is most relevant to our research. First, we present most commonly used local image feature detectors and descriptors. We then review state-of-the-art global representations that encode the distribution of local image descriptors in an image, namely: Bag of Words
Chapter 3 presents several algorithms to compress floating-point, high-dimensional global signatures into compact binary codes including: PCAE, PCAE+RR and PCAE+ITQ. We show that methods advocated by prior-art to reduce quantization error by applying rotation after PCA significantly deteriorate the retrieval performance. To solve this problem we propose a novel matching algorithm for binary descriptors called PCAE with Weighted Hamming distance (PCAE+WH), which effectively minimizes the quantization error introduced by binary conversion. In PCAW+WH, the high energy directions are given more weight by the Hamming distance compared to low energy directions. Finally we show that our binary RVD significantly outperforms any results published up to date, especially on large datasets.

In Chapter 4 we develop a robust and efficient Mobile Visual Search system based on RVD. First, we build a compact, robust and scalable RVD signature with novel Cluster selection and Bit selection mechanisms. Then we show how the compact RVD representations can be effectively and efficiently matched in the compressed domain with weighted Hamming distance. Finally, we compare the retrieval and pairwise matching performance of RVD against Residual Enhanced Visual Vector (REVV) and Scalable Compressed Fisher Vector (SCFV), and we highlight the advantages of RVD over aforementioned representations.

In Chapter 5 we develop an image classification system based on the RVD framework. First, we introduce an effective algorithm which adds second order statistics (diagonal covariance) in the original RVD representation, leading to a significant improvement in performance. Next we present the experimental setup followed by detailed evaluation
and analysis on PASCAL VOC 2007 dataset. Finally, the performance of the proposed approach is comprehensively evaluated on Caltech256 and MIT scene67 datasets.

In Chapter [7] we summarise the major results achieved in this thesis and highlight important lessons learned from the research. We also discuss some possible directions for the future work.

1.5 Publication

Patents

- Compact and robust signature for large scale visual search, retrieval and classification [16]
  Miroslaw Bober, Syed Husain
  Filing date July 7, 2014
  Publication date January 15, 2015
  PCT/GB2014/052058, January 2015

Articles in peer-reviewed journals

- Improving large-scale image retrieval through robust aggregation of local descriptors
  Syed Husain, Miroslaw Bober
  Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)

International peer-reviewed conferences

- Robust and scalable aggregation of local features for ultra large-scale retrieval [40]
  Syed Husain, Miroslaw Bober
  IEEE International Conference on Image Processing, 2014

MPEG Standardisation Contributions
• Improvements to TM6.0 with a Robust Visual Descriptor - Proposal from University of Surrey and Visual Atoms [17]
Miroslaw Bober, Syed Husain, Stavros Paschalakis and Karol Wnukowicz
MPEG Standardisation meeting: ISO/IEC JTC1/SC29/WG11 M30311 CODING OF MOVING PICTURES AND AUDIO, August 2013, Vienna, Austria.

• Improved RVD in TM8 - CE2 Response from University of Surrey and Visual Atoms [18]
Miroslaw Bober, Syed Husain, Stavros Paschalakis and Karol Wnukowicz
MPEG Standardisation meeting: ISO/IEC JTC1/SC29/WG11 M32330 CODING OF MOVING PICTURES AND AUDIO, January 2014, San Jose, US.
Chapter 2

Literature Review

Object recognition and retrieval based on visual appearance are important tasks required in a broad range of applications, including management of multimedia content, mobile commerce, surveillance, or pattern discovery. The task is challenging as the objects to be recognized are often surrounded by background clutter, or partially occluded. Additionally, variations in objects appearance exist due to illumination and view-point changes. Consequently one requires robust techniques that can cope with significant variability of local image measurements. Furthermore, today’s systems must be scalable due to the huge volumes of multimedia data, which can comprise billions of images or video frames.

A classical approach to object based image retrieval involves use of scale-invariant local descriptors such as SIFT [70] or later variants [12], which achieve some robustness to scale changes, illumination conditions and occlusions. While the use of local descriptors increases the robustness against large visual distortions and partial occlusions, it also dramatically increases time of the search, as they need to be individually compared and matched. One solution is to form a single, global image representation, thereby simplifying the matching process and leading to an improved matching speed and lower memory usage.

Figure 2.1 presents a generic pipeline for visual search using global descriptors. Given a query image, the first step is to extract a set of local descriptors. Section 2.1 and
2.1 Keypoints detection

The keypoints can be extracted densely from a regular sampling grid at multiple scales or sparsely from salient regions. These points should be repeatedly detectable in different images that contain the similar object and should also be invariant to scale, translation and rotation. In the following, we describe methods for sparse keypoints...
2.1. Keypoints detection

Moravec [77] created one of the first signal-based keypoint detectors for robotic obstacle avoidance. Harris et al. [37] developed their popular corner detector, which locates corner regions that have large eigenvalues in the second moment matrix. A well-known Hessian detector [13] finds keypoint locations that exhibit high gradients in the two orthogonal directions.

The keypoints extracted by the aforementioned detectors are robust to illumination variations and image rotations [98]. However, the objects to be recognized can appear at different scales, therefore necessitating the use of scale-invariant keypoint detectors. Lindeberg [68] developed a method that searches for scale-space extrema of a scale normalized Laplacian-of-Gaussian (LoG). In [70], Lowe introduced an efficient algorithm for computing extrema in a Difference-of-Gaussian (DoG) scale-space. In [75], Mikolajczyk et al. combined the Hessian detector with the scale selection mechanism by Lindeberg to yield keypoints that provide some level of invariance against scale, illumination change or geometric distortions.

In [11], Baumberg proposed keypoint detector that is invariant to scale, rotation as well as affine viewpoint changes. It is usually implemented as an iterative process where interest points are first detected as extrema in the scale-space. In each iteration, the covariance matrix of gradients of image intensities over the local region is calculated and the eigenvalues of the covariance matrix are computed. A new transformation matrix is learned to make the aforementioned covariance matrix isotropic. The image is then transformed to a new skew normalized frame. Mikolajczyk et al. [74] [75], modified this approach to also re-estimate over scale and position, proposing two popular detectors called Harris-Affine and Hessian-Affine. In [72], Matas et al. proposed to apply watershed segmentation algorithm in order to extract stationary areas of homogeneous intensity for varying threshold values. The set of maximally stable extremal regions (MSER) is invariant to affine intensity changes and projective geometric transformations. A performance comparison of the repeatability of several detectors is provided by Mikolajczyk et al. [76]. Detailed experiments showed that MSER and Hessian-Affine detectors are most repeatable for a variety of imaging conditions.

In [96], Rosten et al. proposed a low complexity detector called Features from Acceler-
2.2. Local descriptors extraction

The next step after keypoint detection involves extraction of robust and discriminative descriptors, representing the image content around the feature points. Lowe \cite{70} developed a very popular and effective Scale Invariant Feature Transform (SIFT), which actually combines a feature point detector and descriptor. It is based on the spatial distribution of pixel intensity gradients in a patch surrounding the keypoint. More precisely, each image patch is first divided into a square grid of $4 \times 4$ spatial cells and a gradient orientation histogram is calculated within each cell using 8 orientation bins, to form $4 \times 4 \times 8 = 128$ dimensional descriptor. The set of histograms is L2-normalized to make it invariant to illumination changes. Colour SIFT \cite{19,115,114} computes SIFT descriptors for each R, G and B colour bands. Ke et al. \cite{57} proposed to reduce the dimensionality of SIFT by projecting the normalized gradient patches via PCA. The SIFT descriptor is truncated to between 12 and 36 dimensions, where 36 dimensional PCA-SIFT produced the best results. Gradient Location and Orientation Histogram (GLOH) descriptor \cite{74} uses log-polar layout for gradient binning, which provided superior repeatability to SIFT but it is more expensive to compute. Bay et al. \cite{12} introduced Speeded Up Robust Features (SURF) descriptor as a computationally effective alternative to SIFT descriptor. The SURF descriptor encodes a distribution of
2.2. Local descriptors extraction

Haar-Wavelet responses within the interest point neighbourhood. Chandrasekhar et al. [20] introduced a low bit-rate descriptor called Compressed Histogram of Gradients (CHoG) by efficiently compressing the gradient distribution in each spatial cell. CHoG has 20 times lower bit rate than SIFT descriptor thus providing significant speed-up for image matching process. Tola et al. [107] developed Daisy descriptor, which uses a circular grid with small overlap between cells and, to improve the descriptor robustness, circular cells of increasing radius, weighted by a Gaussian window. Compared to SIFT, it is much faster to compute for dense point matching and its performance is comparable to SIFT and GLOH. Several modifications were proposed to improve the robustness of the SIFT descriptor: Pyramidal SIFT (P-SIFT) [100] descriptor consists of three SIFT descriptors that describe a patch at multiple level of resolutions, Super-Pixel-based isolation of the SIFT (SP-SIFT) [78], Edge-SIFT [126], Dominant SIFT (D-SIFT) for real time mobile applications [110], compact and robust Binary SIFT (BSIFT) [129] and Nested-SIFT [121]. In [102], Simonyan et al. developed a generic pipeline for training two important components of the local descriptor computation: robust dimensionality reduction and spatial pooling. This has resulted in a very powerful yet complex descriptor.

Recently, local binary descriptors such as BRISK [63], FREAK [84] and BRIGHT [42] are becoming increasingly popular, as they deliver high matching speed, small memory footprint and are significantly faster to compute compared to SIFT and even SURF. The BRISK is an intensity based descriptor, constructed by stacking the output of simple brightness comparison tests. The test pattern defines locations equally spaced on four concentric circles centred at the key-point. Gaussian smoothing is applied to increase the robustness by reducing aliasing. FREAK also uses a circular sampling grid, however the density of points is not constant - it drops exponentially when moving away from the centre, as in the human retina. The main difference with BRISK is the exponential change in sampling density and the overlap in receptive fields. The BRIGHT descriptor is based on a hierarchical Histogram of Gradients (HOG). Compared to BRISK and FREAK, the BRIGHT descriptor is three times more compact and exhibits a similar level of performance in descriptor-by-descriptor matching tests. Conceptually it is also similar to SIFT as it uses histograms of gradients. Other popular binary descriptors
include Ordinal and Spatial Information of Regional Invariants (OSRI) [122], Ultra-
short Binary Descriptor (USB) [125], Binary descriptor based on Asymmetric pairwise
Boosting (Bamboo) [10], Moments based Local Binary Descriptor (MOBIL) [14] and
Binary SIFT (BSIFT) [129].

2.3 Global image representation

This section reviews state-of-the-art global representations that encode the distribution
of local image descriptors in an image, namely the BoW, FV, VLAD, and Triangulation
Embedding.

2.3.1 Bag of Words

The BOW signature has been widely used in textual documents retrieval and subse-
quently adapted for image retrieval by Sivic and Zisserman [104]. BoW is essentially a
histogram, where each local descriptor is assigned to the nearest cluster or visual word.
More precisely, given a set of $T_I$ local descriptors $\{x_1, ..., x_T_I\} \in \mathbb{R}^d$ extracted from a
set of $I$ training images, we learn a visual vocabulary $\{\mu_1, ..., \mu_n\}$ of $n$ cluster centres
via K-means clustering. Given an image, the extracted descriptors are quantized into
a precomputed codebook. To form fixed length $n$-dimensional representation, a his-
togram of descriptors with $n$ bins (visual words) is constructed, where each descriptor
is assigned to the closest (in the Euclidean space) cluster. Figure 2.2 shows how to
form BoW representation of an image.

Several advances were proposed to improve BoW scalability and robustness. Hierarchi-
cal K-means [82] and Approximate K-means clustering [90] algorithms were proposed
to produce large and discriminative vocabularies. The robustness of the BoW system
was improved by using a soft assignment technique [91]. In this approach, each de-
scriptor is assigned to cluster $j$ with assignment weight proportional to $exp(-\frac{dist^2}{2\sigma^2})$,
where $\sigma$ is the spatial scale and $dist$ is distance to the cluster center. Torii et al. [109]
compute recurrent structures in an image and suggest to hard assign the member de-
scriptors, as multiple occurrences of recurrent elements assures fair assignment (natural
Figure 2.2: Bag of Words model. (a) Local descriptors are shown as green circles. (b) The descriptors are clustered using K-means clustering, in order to quantize the space into a finite number of codewords or cluster centres (red circles). (c) For a given query or database image, each extracted descriptor is quantized to its nearest cluster centres. (d) A BoW histogram is computed by counting the number of times each of the cluster centre occurs in an image.

soft-assignment) of descriptors to cluster centres. An alternative approach is based on sparse coding [124], which enforces a descriptor to be assigned to relatively small number of cluster centres. In [117], Wang et al. introduced an effective visual word encoding method called Locality Constrained Linear Coding (LLC), which projects each local descriptor $x_t$ onto its local linear subspace and integrates the projected vectors by max pooling to form the image signature.

In the descriptor matching stage, the BoW signatures are usually weighted using ‘term frequency inverse document frequency’ ($tf$-$idf$) weighting [S2] to reduce the impact of
2.3. Global image representation

non-discriminative visual words. However, the \(tf-idf\) weighting does not take into account the so-called burstiness phenomenon: if a centroid appears once in a particular image, it is more probable that it will appear again in the same image. To solve the aforementioned problem of burstiness, Jegou et al. [49] applied power normalization with a factor of 0.5 on the \(tf\) term of each word. In subsequent work [55] [46], contextual dissimilarity measure is introduced to account for the non-homogeneous spacing of the BoW signatures. Zheng et al. [128] proposed \(L_p\)-norm idf weights to improve performance. In [48], the BoW vectors are projected via PCA and subsequently whitened to limit the impact of visual word co-occurrences [27] (over-counting of some visual patterns when comparing global descriptors). For efficient storage and retrieval, inverted index [82] is used.

Despite all the aforementioned improvements, the performance of the BoW representation still lacks in terms of retrieval accuracy, especially on large scale datasets.

2.3.2 Fisher Vectors

While the BoW is based on the histogram of visual word occurrences in each image, newer methods such as FV or VLAD introduce higher-order statistics.

Fisher Vector (FV) [86] [89] [88] encoding aggregates local image descriptors based on the Fisher Kernel framework [43]. More precisely, let \(X = \{x_t \in \mathbb{R}^d, t = 1...T\}\) be the set of local descriptors, such as SIFT [70] or SURF [12], extracted from an image. Let \(u_\Theta\) be an image-independent probability density function (pdf) which models the generative process of \(X\), where \(\Theta\) represents the parameters of \(u_\Theta\).

Fisher Vector framework (Figure 2.3) assumes \(u_\Theta\) to be a Gaussian Mixture Model (GMM) [86]: \(u_\Theta(x) = \sum_{j=1}^{n} \omega_j u_j(x)\). We represent the parameters of the \(n\)-component GMM by \(\Theta = (\omega_j, \mu_j, \Sigma_j : j = 1...n)\), where \(\omega_j, \mu_j, \Sigma_j\) are respectively the weight, mean vector and covariance matrix of Gaussian \(j\). The covariance matrix of each GMM component \(j\) is assumed to be diagonal and is denoted by \(\sigma_j^2\). The GMM assigns each descriptor \(x_t\) to Gaussian \(j\) with the soft assignment weight \((\tau_{tj})\) given by the
2.3. Global image representation

Figure 2.3: Fisher Vectors. (a) Local descriptors (green circles) are extracted from a set of training images. (b) The parameters of a GMM are learned on the training descriptors using Expectation Maximization algorithm. (c) Given an image, the GMM assigns each descriptor to all Gaussians $j$ (red circles) with a soft assignment weight $\tau_{tj}$. (d) The FV representation $\zeta_{\Theta}$ of an image is obtained by concatenating the gradients $\zeta_j$ for all Gaussians $j = 1..n$.

The posteriori probability:

$$\tau_{tj} = \frac{\omega_j u_j(x_t)}{\sum_{i=1}^{n} \omega_i u_i(x_t)}$$  \hspace{1cm} (2.1)

The $d$-dimensional derivative with respect to mean $\mu_j$ of Gaussian $j$ is denoted by $\zeta_j$:

$$\zeta_j = \frac{1}{T \sqrt{\omega_j}} \sum_{t=1}^{T} \tau_{tj} \frac{x_t - \mu_j}{\sigma_j}$$  \hspace{1cm} (2.2)

The FV representation $\zeta_{\Theta}$ of an image is obtained by concatenating the gradients $\zeta_j$ for all component Gaussian $j = 1..n$ and is therefore $D = d \times n$ dimensional (see Figure...
2.3. Global image representation

Figure 2.4: VLAD. (a) A visual vocabulary of \( n \) cluster centres (red circles) is learned on a training set of local descriptors. (b) Now given an image, each descriptor is quantized to its nearest cluster centres. (c) The residual vectors \( x_t - \mu_j \) are computed between the descriptors and their corresponding cluster centres. (d) The residual vectors in each cluster are accumulated to form the VLAD representation \( \zeta_j \).

2.3.3 VLAD and VLAT

Jegou et al. \cite{54} proposed a simplified version of FV called VLAD. A codebook \( \{\mu_1, ..., \mu_n\} \) of \( n \) cluster centers is obtained via K-means clustering and each descriptor \( x_t \in \mathbb{R}^d \) is hard-assigned to its nearest cluster center \( NN(x_t) \). The main idea here is to compute the cluster level representations \( \zeta_j \in \mathbb{R}^d \) by aggregating the differences \( x_t - \mu_j \).
The final VLAD representation is obtained by concatenating all aggregated vectors \( \zeta_j \) for all clusters \( j = 1, \ldots, n \) and is therefore \( d \times n \) dimensional.

Recently many advancements have been made to the original VLAD representation. Chen et al. [23] investigated different residual vector aggregation methods, namely sum, mean and median. In the sum aggregation approach, the residual vectors in each cluster are accumulated to form vector \( \zeta_j = \sum_{x_t: NN(x_t) = j} x_t - \mu_j \). In the mean aggregation method, the sum of residual vectors is normalized by the number of residual vectors \( (N_j) \) in each cluster \( \zeta_j = \frac{1}{N_j} \sum_{x_t: NN(x_t) = j} x_t - \mu_j \). Median aggregation is similar to mean aggregation, except that the median is computed along each dimension of residual vectors individually. The experimental results on MPEG CDVS dataset [4] show that mean aggregation scheme outperforms sum and median aggregation.

In [8], Arandjelovic et al. introduced two new elements. The first one is called Intra-normalization where the aggregated vectors \( \zeta_j \) are L2 normalized before concatenation (VLAD Intra), in order to reduce the impact of bursty visual elements. The second one proposes extraction of multiple spatial VLAD descriptors from a single image to improve the retrieval of small objects. They also note that the retrieval performance suffers in cases where the visual vocabulary is poorly adjusted to the dataset. To remedy this, they introduced vocabulary adaptation algorithm where the adapted cluster centres \( \hat{\mu}_j \) are computed as a mean of all descriptors in the test dataset that are quantized to cluster \( j \). The aggregated vectors \( \zeta_j \) are recomputed by accumulating residual vectors between descriptors and the adapted centres \( \hat{\mu}_j \).

Dellumeau et al. [28] improved the performance of VLAD by introducing two complementary techniques. The first one applies residual normalization, where each residual vector is L2 normalized before aggregation. The second one attempts to capture large variety of bursty patterns by transforming the residual vectors inside each cluster centre, using a local PCA basis \( P_j \), before aggregation into \( \zeta_j \). The improved version of
VLAD is referred as VLAD\textsubscript{LCS+RN}:

\[ \zeta_j = \sum_{x_t:NN(x_t) = j} P_j^\top \frac{x_t - \mu_j}{||x_t - \mu_j||_2} \]  

(2.4)

In order to capture geometric information, Zhao et al. [127] introduced Covariant-VLAD (CVLAD), constructed by stacking several distinct VLAD vectors. Each VLAD encodes features having the same quantized dominant orientation.

In [120], Xioufis et al. performed a detailed evaluation of the VLAD based image retrieval pipeline. Firstly the impact of employing different types of local descriptors (SIFT, SURF and Colour-SURF) on the retrieval performance was studied. For Colour-SURF (CSURF), the keypoints are extracted using the original SURF method on the grey-scale image but a SURF descriptor is computed for each RGB color band. The CSURF descriptor is formed by stacking the three SURF descriptors. The experimental results show that a VLAD+CSURF and VLAD+SURF outperforms a VLAD+SIFT pipeline. Secondly, a comparison between two residual vector aggregation methods, sum and mean aggregation was performed showing that sum aggregation provides superior retrieval performance. Thirdly, the dimensionality of VLAD+SURF descriptor is reduced to 128-dimensions via PCA followed by Whitening operation.

Christian et al. [31] suggested to apply cluster-wise PCA on aggregated residual vectors before concatenation and named their representation HVLAD:

\[ \zeta_j = P_j^\top \sum_{x_t:NN(x_t) = j} \frac{x_t - \mu_j}{||x_t - \mu_j||_2} \]  

(2.5)

In [65], Li et al. introduced a modified version of VLAD named as pVLAD where \(l_p\)-norm IDF weights are applied to each residual vectors before aggregation into the cluster level representation.

In [69], Liu et al. proposed a Hierarchical VLAD (HiVLAD), where the cluster centres \(\mu_j\) obtained from the K-means clustering are further divided into \(g\) sub-clusters \(\{\mu^i_j|i = 1, 2, ..., g\}\). For each sub-cluster, the residual vectors \(x_t - \mu^i_j\) are computed and subsequently L2-normalized. The residuals are accumulated into sub-cluster level representation \(\zeta^i_j\):

\[ \zeta^i_j = \sum_{x_t:NN(x_t) = i} \frac{x_t - \mu^i_j}{||x_t - \mu^i_j||_2} \]  

(2.6)
The $\zeta_j^i$ vectors are aggregated to form cluster level representation $\zeta_j$ and finally the $\zeta_j$ vector is L2-normalized to improve the discriminability of HiVLAD:

$$\zeta_j = \frac{\sum_{i=1}^{g} \zeta_j^i}{\| \sum_{i=1}^{g} \zeta_j^i \|_2}$$  \:(2.7)

This approach outperforms the original VLAD as the local descriptors are better centred resulting in a more uniform distribution of residual vectors.

Picard et al. [92] [79], introduced Vector of Locally Aggregated Tensors (VLAT) descriptor, formed by aggregating tensor products of descriptors. As in VLAD, a visual vocabulary $\{\mu_1, ..., \mu_n\}$ of $n$ cluster centres is learned and the covariance matrix $\Sigma_j$ of descriptors belonging to cluster $j$ is computed as:

$$\Sigma_j = \frac{1}{N_j} \sum_{i=1}^{I} \sum_{x_t:NN(x_t)=j} (x_t - \mu_j)(x_t - \mu_j)^\top$$  \:(2.8)

where $N_j$ is the total number of descriptors in cluster $j$ for the training sample set of $I$ images. A cluster level representation $\zeta_j$ is computed by accumulating the centred tensors of the centred descriptors for each cluster $j$:

$$\zeta_j = \sum_{x_t:NN(x_t)=j} (x_t - \mu_j)(x_t - \mu_j)^\top - \Sigma_j$$  \:(2.9)

Each $\zeta_j$ is symmetric, thus the upper triangles and diagonals are extracted and unfolded to form the VLAT vector. In [80], Negrel et al. improved the efficiency and accuracy of the original VLAT representation by proposing two extensions: (i) PCA cluster-wise VLAT (PVLAT), which applies PCA to each flattened cluster representation and (ii) Compression of PVLAT vector (CPVLAT) using PCA matrix

### 2.3.4 Triangulation embedding (Temb)

Recently, Jegou et al. [47] proposed an aggregation scheme using triangulation embedding (illustrated in Figure 2.5). In this approach, K-means clustering is used to learn a codebook $\{\mu_1, ..., \mu_n\}$ of $n$ visual words and each descriptor $x_t$ is hard-assigned to all cluster centres. The residual vectors $x_t - \mu_j$ are computed and subsequently L2 normalized to yield a set $(s_{tj})$ of normalized residual vectors:

$$s_{tj} = \left\{ \frac{x_t - \mu_j}{\|x_t - \mu_j\|_2} \right\} \text{ for } j = 1...n,$$  \:(2.10)
2.3. Global image representation

\[ \mu_1 \mu_3 \mu_4 \mu_2 \]

\[ \mu_5 \]

\[ \mu_1 \mu_3 \mu_4 \mu_2 \mu_5 \]

\[ s_{12} s_{11} s_{14} s_{15} \]

\[ (c) (d) \]

\[ \phi_{\Delta t} = \Sigma^{-1/2}(S_t - S_0) \] (2.11)

where \( S_0 \) and \( \Sigma \) are respectively the expected value and covariance matrix associated with \( S \). The \( \phi_{\Delta t} \) vectors are aggregated using sum aggregation (equation 2.12 and 2.13).
to form a global image representation $\psi_s$:

$$\psi_s(X) = \sum_{x_t \in X} \phi_{\Delta t}$$

$$= \Sigma^{-1/2} \left( \sum_{x_t \in X} S_t \right) - T \Sigma^{-1/2} S_0$$

The drawback of sum aggregation is that the aggregated vector $\psi_s$ is more influenced by common uninformative local descriptors rather than rare but informative ones. This problem is alleviated by aggregating descriptors $\phi_{\Delta t}$ using democratic aggregation where a weight $\varphi_t$ is applied to each $\phi_{\Delta t}$ before aggregation into global signature $\psi_d$. The weights $\varphi_t$ are learned using modified Sinkhorn algorithm [58]:

$$\psi_d(X) = \sum_{x_t \in X} \varphi_t \phi_{\Delta t}$$

The global descriptors $\psi_s$ and $\psi_d$ are rotated with a PCA matrix and then power-law normalization is applied to obtain the final representations.

### 2.3.5 Global descriptors based on Convolutional Neural Networks (CNN)

Recent research has shown that image descriptors computed using deep CNNs achieve state-of-the-art performance for image retrieval and classification tasks. The CNN-based descriptors can be aggregated to form a robust and discriminative global image signature [9] [81].

In [81], Ng et al. proposed to encode CNN-based descriptors into a VLAD representation. More precisely, an RGB image is first resized into a $c \times c$ square and a mean RGB value is subtracted from each pixel. The image is then passed through a pre-trained network comprising of $L$ convolutional layers. The state-of-the-art CNN used for experiments are OxfordNet [103] and GoogLeNet [105]. The output of a $l$-th layer $L^l$ is a $c^l \times c^l \times d^l$ feature map, where $d^l$ is the number of filters corresponding to $L^l$. A set $X^l = \{x^l_{1,1}, x^l_{1,2}, ..., x^l_{c^l,c^l}\}$ of $d^l$-dimensional feature vectors is obtained at each location $(a, b)$, $1 \leq a \leq c^l$ and $1 \leq b \leq c^l$, in the feature map. As in the SIFT-based approaches,
2.4 Compression of global descriptors

A codebook \( \{ \mu_1^l, ..., \mu_n^l \} \) of \( n \) cluster centres is learned using a set of training images. For each centroid, the residual vectors \( x_{a,b}^l - \mu_j^l \) are accumulated to obtain a VLAD signature \( \zeta_j^l \) regarding layer \( L^l \):

\[
\zeta_j^l = \sum_{x_{a,b}^l: \text{NN}(x_{a,b}^l) = j} x_{a,b}^l - \mu_j^l
\] (2.15)

The VLAD+CNN representation is obtained by concatenating all aggregated vectors \( \zeta_j^l \) for all \( n \) visual words. The experimental results on Holidays and Oxford dataset showed that the best performing layers are Inception 3a, 5b and 4e on GoogLeNet, and conv 4-2, 5-1 and 5-2 on OxfordNet respectively.

Babenko et al. [9] aggregated deep convolutional descriptors to form global image representations: FV, Temb and SPoC. The SPoC signature is obtained by applying sum-pooling function to the weighted descriptors. Larger weights are applied to descriptors from the centre of the feature map. The SPoC descriptor outperforms FV and Temb representation on Holidays and Oxford datasets.

The aforementioned global signatures are high dimensional which makes them unsuitable for large scale retrieval tasks. In the next section, we will review state-of-the-art algorithms to compress global image representations.

2.4 Compression of global descriptors

The global descriptor size, expressed as bytes per image, has a significant impact on the performance of an image retrieval system; ideally the descriptors for entire database should fit in the RAM memory of the server for fast processing. Recently several notable techniques have been introduced to compress floating-point global image descriptors into compact codes.

In [45], Jegou et al. applied PCA to reduce the dimensionality of BoW signatures. In [44], two methods of converting BoW vector to binary code are presented. The first method simply assigns 1 if a particular centroid is present in the image, 0 otherwise. The second method takes a high-dimensional and sparse BoW vector and projects it
2.4. Compression of global descriptors

using vocabulary aggregators into several dense and shorter miniBoW vectors. Binary signatures are generated for each miniBoW vector using Hamming Embedding method \[51\] and distance between signatures is calculated using Hamming distance. This method achieves good retrieval results on Holidays datasets, with significantly reduced memory usage. Chum et al. \[26\] \[25\] \[24\] compute a compact image representation by applying the MiniHash algorithm to the BoW vectors. Jegou and Chum \[48\] improved the retrieval performance by stacking multiple BoW signatures for an image followed by PCA transformation and whitening operation.

In \[48\], Jegou et al. proposed a successful approach to dimensionality reduction of VLAD and FV signatures. It computes multiple VLAD descriptors using distinct visual vocabularies, concatenates them and applies dimensionality reduction using PCA and whitening. The compact global descriptors are respectively indexed and searched accurately and efficiently using Product Quantization (PQ) and the IVFADC method \[54\].

Perronnin et al. \[88\] compressed high dimensional FV into binary codes using three methods: Sign Binarization, Local Sensitive Hashing \[119\] and Spectral Hashing \[6\]. In the Sign Binarization approach, a value 1 is assigned to any positive coefficient of \(\zeta_\Theta\) while a value 0 is assigned to any negative coefficient. Local Sensitive Hashing is popular binary encoding method in which the global descriptor is first projected using a Gaussian random matrix and the sign binarization is performed. The Spectral Hashing algorithm seeks balanced and uncorrelated binary signatures of data points by applying the following steps: (1) Compute the eigenvectors and associated eigenvalues of the data using PCA, (2) Calculate the Laplacian eigenfunctions along every PCA direction of the data, and (3) Quantize the Laplacian eigenfunctions at zero, to produce binary signatures. For all methods, the similarity between two binary \(\zeta_\Theta\) signatures is computed using standard Hamming distance.

Gordo et al. \[35\] applied PCAE, PCAE+RR and PCAE+ITQ methods to covert FV into binary codes. In the PCAE approach, the global descriptor is projected via a PCA matrix and sign binarization is performed. Jegou et al \[54\] suggested to apply a random rotation to the PCA-transformed data (PCAE+RR) in order to balance the energy of
2.4. Compression of global descriptors

Different PCA directions. Gong et al. [34] proposed an approach called iterative quantization (ITQ) for refining the random rotation matrix to reduce the quantization error. The projected vector is converted to compact signature using sign binarization method. In [35], two methods are used to compute similarity between binary codes: Hamming distance and Asymmetric distance. Experimental results presented in [35] show that PCAE+ITQ provides the best performance on CIFAR, UKB and Holidays datasets (detailed information about the datasets is presented in Section 3.4.1). Furthermore, the results on Holidays dataset show that computing similarity between two binary codes using Asymmetric distance provides significant gain in retrieval accuracy (+4%) compared to standard Hamming distance. However, the Asymmetric distance is computationally more complex (in-terms of matching time) as the query is not converted into binary code.

In [80], Negrel et al. converted high dimensional VLAT signatures into binary codes by projecting them onto a lower dimensional space with a PCA matrix, and then applying sign binarization.

Chen et al. [23] introduced Residual Enhanced Visual Vectors (REVV), where VLAD global descriptor is first reduced using linear discriminative analysis (LDA). More precisely, let $Y = \{(y_{1,1}, y_{1,2}), \ldots (y_{J,1}, y_{J,2})\}$ denotes a set of $J$ matching image pairs and $Z = \{(z_{1,1}, z_{1,2}), \ldots (z_{J,1}, z_{J,2})\}$ represents a set of $J$ non-matching image pairs. To find the LDA transform matrix, the following problem is solved, independently for each centroid:

$$
\text{maximize} \sum_{z_1, z_2 \in Z} (v_j ^\top (\zeta_{z_1,j} - \zeta_{z_2,j}))^2 \\
\sum_{y_1, y_2 \in Y} (v_j ^\top (\zeta_{y_1,j} - \zeta_{y_2,j}))^2
$$

(2.16)

where $\zeta_j$ is the VLAD representation for the $j$-th cluster centre. The objective in Equation 2.16 is to maximize the ratio of inter class variance to intra class variance, over the projection direction $v_j$. The solution is determined by the eigenvector $v_{j,1}$ of two scatter matrices, associated with the largest eigenvalue $\lambda_{j,1}$:

$$
\Delta_j = \sum_{y_1, y_2 \in Y} (\zeta_{y_1,j} - \zeta_{y_2,j})(\zeta_{y_1,j} - \zeta_{y_2,j})^\top
$$

(2.17)

$$
\Lambda_j = \sum_{z_1, z_2 \in Z} (\zeta_{z_1,j} - \zeta_{z_2,j})(\zeta_{z_1,j} - \zeta_{z_2,j})^\top
$$

(2.18)
2.4. Compression of global descriptors

\[ \Lambda_j v_{j,1} = \lambda_{j,1} \Delta_j v_{j,1} \]  
(2.19)

The transform matrix \( P_{lda,j} \in \mathbb{R}^{d_{lda} \times d'} \) for each cluster \( j \) consists of \( d_{lda} \) generalized eigenvectors corresponding to the \( d_{lda} \) largest eigenvalues:

\[ P_{lda,j} = [v_{j,1}...v_{j,d_{lda}}] \]  
(2.20)

The dimensionality of the aggregated vector \( \zeta_j \) is reduced from \( d' \) to \( d_{lda} \) dimension using matrix \( P_{lda,j} \). The binary signature \( \zeta_j^b \) is obtained by keeping the sign of each component of the transformed vector:

\[ \zeta_j^{lda} = P_{lda,j} \zeta_j \]  
(2.21)

\[ \zeta_j^b = [\text{sign}\{(\zeta_j^{lda})_1\}; \text{sign}\{(\zeta_j^{lda})_2\};...;\text{sign}\{(\zeta_j^{lda})_{d_{lda}}\}] \]  
(2.22)

The binary REVV representation is obtained by stacking all vectors \( \zeta_j^b \) for \( j = 1,..,n \) and is \( d_{lda} \times n \) dimensional. The similarity between two REVV descriptors is computed using standard Hamming distance.

In [67] [30] [66], a compact global image representation based on Fisher Kernel framework is developed named as Scalable Compressed Fisher Vector (SCFV). The SCFV is computed by exploiting the property of rich sparseness integral to the FV. During the global descriptor formation process, a measure of the aggregated contribution received by the \( j^{th} \) cluster is denoted by the quantity \( \tau_j = \sum_{t=1}^{T} \tau_{tj} \). When \( \tau_j \) is low, the cluster centre \( \mu_j \) is positioned far from most of the image descriptors, and the residual vectors \( x_t - \mu_j \) are therefore less discriminative. Hence, the aggregated vector \( \zeta_j \) for the \( j^{th} \) codeword is discarded if \( \tau_j \) is below a threshold value \( \tau_{th} \). This threshold value can be used to control size of the descriptor. The scalable FV is computed as follows:

\[ \zeta_j = \frac{\delta_j}{T\sqrt{\omega_j}} \sum_{t=1}^{T} \tau_{tj} \frac{x_t - \mu_j}{\sigma_j} \]  
(2.23)

\[ \delta_j = \begin{cases} 
1, & \text{if } \tau_j > \tau_{th} \\
0 & \text{otherwise} 
\end{cases} \]  
(2.24)
The SCFV representation $\zeta_{\Theta}$ of an image is formed by stacking the vectors $\zeta_j$ for each of the $j^{th}$ Gaussian in the GMM. The SCFV $\zeta_{\Theta}$ is binarized by employing a sign function. The similarity between two SCFV descriptors is computed using Hamming distance.

In the following section we will compare the retrieval performance of the aforementioned global image representation on the Holidays, Oxford5k and UKB datasets.

2.5 Comparison of global descriptors

The top part of Table 2.1 demonstrates the performance of global image representations formed by aggregating 'shallow hand-crafted' local descriptors.

Compared to the BoW, which only records the count of local descriptors in each visual word, the FV encodes the higher order statistics, resulting in a more dense and discriminative representation and hence better performance. Even though FV is typically computed using a relatively small visual vocabulary of between 64 and 256 words, its retrieval accuracy outperforms that of BoW even when a large vocabulary with 1M - 16M words is used. There are two main drawbacks of aggregating local descriptors in to a FV representation. Firstly, local descriptors actually rarely follow GMM distribution in the feature space, impacting negatively the model performance. Secondly, training of generic GMMs is difficult, especially in high dimensional spaces, and frequently results in sub-optimal solutions, due to the optimisation process being stuck in local minima. This motivates us to develop the RVD framework presented in Chapter 3.

The performance of original VLAD descriptor is lower than the FV on all datasets. However, the most recent HiVLAD offers a gain of +23% and +11.5% in terms of mean Average Precision (mAP) on the Holidays and Oxford5k datasets, compared to FV. The main drawback of VLAD is the limited robustness to outliers. A single descriptor located far from the cluster centre can outweigh the combined contribution from many other vectors located close to that centre.

Experimental results presented in [47] show that Triangulation Embedding performs very well on all standard benchmarks, with $\phi_\Delta + \psi_d$ considerably outperforming FV, HiVLAD and $\phi_\Delta + \psi_s$. However, the extraction time of $\phi_\Delta + \psi_d$ is typically two orders
Table 2.1: Comparison of global image representations on Oxford5k, Holidays and UKB datasets. All results are presented in terms of mAP(%) except for recall@4 for UKB.

<table>
<thead>
<tr>
<th>Method</th>
<th>Oxford5k</th>
<th>Holidays</th>
<th>UKB</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoW [54]</td>
<td>36.4</td>
<td>54.0</td>
<td>2.81</td>
</tr>
<tr>
<td>FV [54]</td>
<td>41.8</td>
<td>60.7</td>
<td>3.35</td>
</tr>
<tr>
<td>VLAD [54]</td>
<td>37.8</td>
<td>55.6</td>
<td>3.28</td>
</tr>
<tr>
<td>VLAD Intra [8]</td>
<td>55.8</td>
<td>65.3</td>
<td>-</td>
</tr>
<tr>
<td>VLAD_{LCS+RN} [28]</td>
<td>51.7</td>
<td>65.8</td>
<td>-</td>
</tr>
<tr>
<td>PVLAT [80]</td>
<td>54.2</td>
<td>66.4</td>
<td>-</td>
</tr>
<tr>
<td>CPVLAT [80]</td>
<td>-</td>
<td>70.0</td>
<td>-</td>
</tr>
<tr>
<td>HVLAD [31]</td>
<td>47.2</td>
<td>69.1</td>
<td>-</td>
</tr>
<tr>
<td>VLAD+SURF [120]</td>
<td>32.8</td>
<td>64.9</td>
<td>3.20</td>
</tr>
<tr>
<td>VLAD+CSURF [120]</td>
<td>-</td>
<td>71.7</td>
<td>3.52</td>
</tr>
<tr>
<td>HiVLAD [69]</td>
<td>63.8</td>
<td>72.1</td>
<td>3.56</td>
</tr>
<tr>
<td>$\phi_{\Delta} + \psi_d$ [47]</td>
<td>67.6</td>
<td>77.1</td>
<td>-</td>
</tr>
<tr>
<td>$\phi_{\Delta} + \psi_s$ [47]</td>
<td>63.3</td>
<td>74.5</td>
<td>-</td>
</tr>
<tr>
<td>VLAD+CNN [81]</td>
<td>64.9</td>
<td>83.8</td>
<td>-</td>
</tr>
<tr>
<td>SPoC+CNN [9]</td>
<td>65.7</td>
<td>80.2</td>
<td>-</td>
</tr>
</tbody>
</table>

of magnitude higher than FV and $\phi_{\Delta} + \psi_s$. The overall extraction time is prohibitively high, in fact it prevented the authors from performing experiments on datasets larger than 100K using 64 cluster centres. Therefore [47] uses a small codebook ($n = 16$) to perform experiments on Holidays1M dataset, resulting in retrieval accuracy below that achieved by the state-of-the-art global representations (refer Table 3.4). Also, as we show later in the experimental Section 3.4, $\phi_{\Delta} + \psi_d$ performance deteriorates rapidly when the data set increases beyond 1M, particularly when the descriptor dimensionality is reduced, making it unsuitable for large scale retrieval.

The performance of global signatures, formed by aggregating deep convolutional features, is presented in the bottom part of Table 2.1. It can be observed that CNN-based representations provide a significant gain in retrieval performance especially on Holi-
days dataset. However, the focus of this thesis is to aggregate shallow hand-crafted descriptors into a robust global descriptor.

In the next Chapters, we aim to address the aforementioned drawbacks of state-of-the-art global representations and present our novel Robust Visual Descriptor.
Chapter 3

Robust Visual Descriptor

In this chapter we introduce the Robust Visual Descriptor (RVD), a generic local descriptor aggregation framework to underpin tasks of image retrieval and classification. Our RVD descriptor is inspired by concepts from Robust Statistics [39]. In retrieval, image pairs with matching visual objects contain a certain proportion of matching local descriptors, contaminated by a large proportion of non-matching outliers. For example, in the Oxford dataset, the median percentage of inliers is only 20%. Thus the task of image matching may be considered as detection of matching local descriptor pairs in the strong presence of outliers. The task is therefore to develop a global representation of a set of local descriptors that will be both representative and robust in the mathematical sense, i.e. not affected by a large number of additional local descriptors.

Our main contributions, as highlighted in Figure 3.1, include:

- We propose a robust global image representation (Robust Visual Descriptor (RVD)), with rank-based multi-assignment of local descriptors and direction-based aggregation achieved by the use of L1-norm on residual vectors. We conduct a thorough experimental study to illustrate the effects of various elements of the RVD pipeline in order to understand the factors supporting its superior performance. For instance, we analyse the behaviour of the rank-based multi-assignment and compare it to the well studied hard-assignment of VLAD and the soft-assignment used in the FV approach.
We further improve the retrieval performance of the RVD method by balancing the variance of residual vector directions in order to maximize the discriminatory power of the aggregated vectors. This is achieved by introducing a novel intra stage pre-processing of the residual directions using cluster-wise PCA with a whitening operation. We call this representation as RVD-W.

In order to neutralize the impact of visual word co-occurrences and increase the discriminatory power of our representation, we propose a new normalization approach applied after the RVD-W vectors are transformed via global PCA. Our normalization involves L1-norm followed by a power-norm (parametrized by factor $\beta$). This new approach benefits not only RVD-W but also FV and VLAD representations.

To work with very large databases, we employ Product Quantization (PQ) to RVD-W descriptor.

This chapter is organized as follows: Section 3.1 presents the RVD descriptor which combines rank-based multiple assignment with the robust aggregation framework. In Section 3.2 we present two improved versions of RVD representation namely RVD with Local PCA (RVD-P) and RVD with Local Whitening (RVD-W). In Section 3.2.3 we propose a novel global descriptor normalization method in order to reduce the impact of visual word co-occurrences. The compression of local descriptor into compact code using Optimized Product Quantization (OPQ) is described in Section 3.3. The
3.1. Rank-based multiple assignment

RVD is a global image representation formed by robustly aggregating local descriptors. In this approach every descriptor is defined by its relative position with respect to a set of reference points (cluster centres) in a \(d\)-dimensional space. More precisely, the \(n\) cluster centres, or the codebook \(\{\mu_1, ..., \mu_n\}\) is computed and each descriptor \(x_t\) is assigned to its \(K\)-nearest clusters \((\text{NN}_K^\gamma)\), where \(\{\gamma = 1...K\}\) denotes the rank of a particular nearest cluster. In the following paragraph we will discuss several strategies of assigning descriptors to visual words and present our novel rank-based multiple assignment.

1) Single assignment (SA): Each local descriptor \(x_t\) is assigned to its nearest cluster \((K=1)\) with assignment weights 1. This is also known as hard assignment. We introduce the following notation: for descriptor \(x_t\), \(\text{NN}_1^\gamma(x_t)\) returns the cluster index that is rank \(\gamma\) from \(x_t\). In case of SA where only one nearest neighbour centre is assigned to each descriptor, we can specify assignment weight as \(\tau_{tj} = 1\) if \(\text{NN}_1^\gamma(x_t) = j\) and \(\tau_{tj} = 0\) otherwise. The drawback of single assignment is that it leads to high quantization error (i.e. matching descriptors are assigned to different clusters) because of the inherent variability in the extracted descriptors. Also the population of vectors assigned to each cluster is small, which is not desirable for robust statistical processing.

2) Multiple assignment (MA): The aforementioned quantization error can be reduced by assigning descriptors to multiple clusters (typically \(K=2, 3\)) with constant assignment weight \(\tau_{tj} = 1\) if \(\text{NN}_1^\gamma(x_t) = j\) and \(\tau_{tj} = 0\) otherwise. However this approach doesn’t take into account that local descriptors belonging to neighbourhoods...
with lower ranks are typically more reliable. The term reliability means that the two matching descriptors (inliers) with specific ranks $\gamma$ are assigned to same cluster centre.

3) **Soft assignment (SoftA):** In this method each descriptor $x_t$ is assigned to cluster $j$ with the soft assignment weight ($\tau_{tj}$) given by the posteriori probability:

$$
\tau_{tj} = \frac{\exp\left(-\frac{1}{2}(x_t - \mu_j)^T \Sigma_j^{-1}(x_t - \mu_j)\right)}{\sum_{i=1}^{n} \exp\left(-\frac{1}{2}(x_t - \mu_i)^T \Sigma_i^{-1}(x_t - \mu_i)\right)} \tag{3.1}
$$

where $\mu_j$ and $\Sigma_j$ are respectively the mean vector and diagonal covariance matrix of cluster $j$.

While SoftA has been shown to deliver superior results and is generally considered the state-of-the-art, little studies exist in the local descriptor assignment pattern and behaviour. We demonstrate later in this section that SoftA has a significant weakness and can be improved in the context of aggregation schemes. One of them is that SoftA often (60% of all the cases) degrades to single assignment. Another one is that the assignment weights depend on the distances between a descriptor and cluster centres, meaning that the contributions from various descriptors are unbalanced and noise in descriptor values impacts the assignment weights. These observations motivate us to introduce the rank-based multiple assignment approach.

4) **Rank-based multiple assignment (RankA):** The rank-based multiple assignment scheme aims to address the aforementioned drawbacks of the SA, MA and SoftA methods. Firstly, it reduces the assignment error by effectively quantizing descriptors to multiple cluster centres. Secondly, it increases the probability that many clusters have a sizeable population of local descriptors assigned to them. Finally, the descriptors are assigned to $K$-nearest clusters with stable rank weights leading to a more balanced and reliable global image representation as compared to the MA and SoftA approaches.

In the proposed RankA, each local descriptor $x_t$ is quantized to $K$-nearest cluster centres ($K=3$ was found to be optimum Fig.3.3(e)) and the assignment weights used for aggregation are derived from ranks. We define assignment weights based on the empirical probability that two descriptors forming a matching pair (inliers) with specific rank are assigned to the same cluster. This probability strongly depends on the proximity
Figure 3.2: Impact of rank-based weighting on performance of (a) Holidays dataset (b) Oxford5k dataset

of the descriptors to the cluster centre in feature space, which can be approximated by the assignment rank $\gamma$. We expect rank one assignments to be more stable than rank three.

Our procedure to find optimal weights include two steps: (1) finding matching descriptor pairs and (2) estimating probabilities of the aligned cluster assignment, given the assignment rank. Detailed procedure of computing assignment weights for each rank $\gamma$ is as follows:

- A matching image pair is selected from the training dataset and SIFT descriptors are extracted from each image. A set of putative matches is computed between the matching pair. Each putative match comprises of a pair of SIFT descriptors, one in each image, that pass Lowe’s [70] ratio test. For a descriptor $x$ in query image, if $NN_1(x)$ and $NN_2(x)$ are the first and second nearest neighbours descriptors in reference image, then the correspondence $(x,NN_1(x))$ is considered a true match if and only if $dist(x,NN_1(x)) \leq th \times dist(x,NN_2(x))$ for some threshold $0 < th < 1$. We experimentally found that threshold 0.8 worked well for SIFT descriptors.

- The RANSAC algorithm is applied on the putative SIFT matches to estimate an affine transformation together with the set of inlier point pairs ($Y$) consistent
3.1. Rank-based multiple assignment

Figure 3.3: Fisher Vectors and RVD statistics on Holidays dataset. (a) Probability distribution for number of nearest clusters \((K)\) that a descriptor is assigned to in FV, (b) Distribution of \(NN_1^K\) soft assignment weights in FV. About 30% of descriptors are assigned with soft assignment weight of 1, (c) Distribution of \(NN_2^K\) soft assignment weights in FV, (d) Rank assignment weights used in RVD encoding. In RankA, each descriptor \(x_t\) is assigned to three nearest clusters, \(NN_1^K\), \(NN_2^K\) and \(NN_3^K\), with assignments weights equal to 1, 0.5 and 0.25 respectively, (e) Performance of RVD as a function of maximum numbers of assigned clusters, (f) Performance of FV as a function of maximum numbers of assigned clusters. The size of codebook is 128.
3.1. Rank-based multiple assignment

with that transform.

- Given an inlier point pair \( y_i, y_j \in Y \), we calculate the probability (\( \Omega_\gamma \)) such that
\[
NN^K_\gamma(y_i) = NN^K_\gamma(y_j),
\]
for \( \gamma = 1, 3 \). The assignment weights for each \( \gamma \) are calculated as: \( \Omega_1 / \Omega_\gamma \). We computed experimentally that in \( NN^K_1 \), \( NN^K_2 \) and \( NN^K_3 \), the probability that an inlier point pair is assigned to the same cluster centre is approximately 0.58, 0.28 and 0.14, therefore assignment weights used in RVD aggregation are:
\[
\tau_{ij} = 1 \text{ if } NN^K_1(x_t) = j, \quad \tau_{ij} = 0.5 \text{ if } NN^K_2(x_t) = j, \quad \tau_{ij} = 0.25 \text{ if } NN^K_3(x_t) = j \text{ and } \tau_{ij} = 0 \text{ otherwise.}
\]

We performed experiments to show the advantage of neighbourhood rank weighting in the RVD aggregation process. It can be observed in Figure 3.2 that weighted rank level combination gives an average improvement of 1.3% and 1.5% on the retrieval accuracy of Holidays and Oxford5k datasets compared to rank level combination with equal weights:
\[
\tau_{ij} = 1 \text{ if } NN^K(x_t) = j \text{ and } \tau_{ij} = 0 \text{ otherwise.}
\]

Analysis of Soft Assignment and Rank-based multiple assignment

In order to better understand the difference between soft assignment (as employed in FV) and our rank-based assignment, we computed FVs for images in the Holidays dataset and analysed cluster assignment statistics and behaviour. The following observations are made. Figure 3.3(a) shows a discrete probability distribution of number of nearest clusters (\( K \)) that a descriptor is assigned to with soft assignment weight greater than 0.1. It can be observed that in 60% of all the cases the weight assignment in FV is such that only the first nearest cluster has a weight exceeding 0.1. This effectively means that SoftA frequently degrades to single assignment. In RankA, a descriptor is always assigned to three nearest centroids. Figure 3.3(b) and Figure 3.3(c) show the distribution of soft assignment weights corresponding to \( NN^K_1 \) and \( NN^K_2 \) respectively and it can be seen that in \( NN^K_1 \) about 30% of descriptors are assigned with soft assignment weight of 1. In RankA, each descriptor \( x_t \) is assigned to three nearest clusters, \( NN^K_1 \), \( NN^K_2 \) and \( NN^K_3 \), with assignments weights equal to 1, 0.5 and 0.25 respectively as shown in Figure 3.3(d). In Fisher vector encoding, many assignment weights \( \tau_{ij} \) are likely to be very small or negligible. We evaluate the performance of FV on the Holidays
3.1. Rank-based multiple assignment

Figure 3.4: Impact of rank-based aggregation on performance for (a) Holidays, (b) Oxford5k, (c) UKB, (d) Holidays1M and (e) Oxford1M (all results in mAP(%) except for recall@4 for UKB)
dataset by setting to zero all but the $K$-largest assignments for each input descriptor $x_t$. Figure 3.3(f) shows that there is no significant change in performance for $K > 3$. This means that there are no benefits from using more than three nearest neighbours.

We evaluate the performance of our rank based multiple assignment (RankA) used in RVD, single assignment (SA) employed in VLAD, and soft assignment (SoftA) employed in FV, as a function of global descriptor dimensionality $D$. To ensure a fair comparison between different methods, all the parameters including SIFT dimensionality and vocabulary size are kept the same. More detail about the experimental set-up is presented in Section 3.4. It can be observed from Figure 3.4 that RankA performs better than SA and SoftA approaches on all datasets. Compared to SoftA, the retrieval accuracy obtained using RankA is significantly higher on large scale datasets, resulting in an average gain of 3.6% and 1.9% on Holidays1M and Oxford1M respectively.

### 3.1.1 Direction preserving mapping function

Figure 3.5 shows the distribution of L1-norms of residual vectors, where it can be seen that the contribution of individual descriptors to cluster level representation varies significantly. We noticed that aggregating non-normalized residual vectors leads to suboptimal performance as the cluster level representations can be strongly influenced by outliers with higher magnitudes of residual errors.
The aforementioned problem is illustrated in Figure 3.6. The dotted polygons indicate Voronoi cells. The cluster centres and the local descriptors are represented by red triangles and green circles respectively. In Figure 3.6(a), the descriptors $x_1$, $x_2$ and $x_3$ are assigned to their nearest cluster centre $\mu_1$, and the corresponding residual vectors, $(x_1 - \mu_1)$, $(x_2 - \mu_1)$, $(x_3 - \mu_1)$ are aggregated to form cluster-level representation $\zeta_1^1$ of query image. Similarly, the aggregated vector of matching image is represented as $\zeta_2^1$ (Figure 3.6(b)). The Euclidean distance between $\zeta_1^1$ and $\zeta_2^1$ is very small. Now assume that due to cluster assignment error, an outlier descriptor $x_4$ is assigned to $\mu_1$ (Figure 3.6(c)). The cluster level representation of query image is strongly influenced by outlier and the Euclidean distance between $\zeta_1^1$ and $\zeta_2^1$ becomes significantly high.

To alleviate this problem, we propose that RVD aggregation encodes each local descriptor using only direction, discarding magnitude. More precisely, for each local descriptor and the associated clusters $\mu_j$ with ranks $\gamma$, the residual vectors $x_t - \mu_j$ are L1-normalized before aggregation. This direction preserving mechanism limits the impact of outliers that happen to be located far from cluster centre. In effect the influence of any single descriptor on the aggregated representative value is now limited and similar in impact for all descriptors.

Figure 3.7 shows the benefit of applying L1-normalization to residual vectors before aggregation. The use of L1-normalization brings an average gain of 1.1% and 1.3% in mAP on the Holidays and Oxford5k datasets respectively.

**RVD formation**

Each normalized residual vector belonging to cluster $j$ is weighted based on rank assignment weights $\tau_{tj}$ to yield vector $r_{tj}$:

$$r_{tj} = \tau_{tj} \frac{x_t - \mu_j}{||x_t - \mu_j||_1}$$

(3.2)

The cluster level representation $\zeta_j$ is computed by aggregating vectors $r_{tj}$ across all ranks $\gamma$:

$$\zeta_j = \sum_{\gamma=1}^{K} \sum_{x_t:NN^K(x_t)=j} r_{tj}$$

(3.3)
3.1. Rank-based multiple assignment

Each $\zeta_j$ is L2-normalized (intra-normalization) in order to make each aggregated vector $\zeta_j$ contribute equally to the final RVD representation $R$. The dimensionality of vector $R$ is $D = d \times n$:

$$R = \begin{bmatrix} \frac{\zeta_1}{\|\zeta_1\|_2} & \frac{\zeta_2}{\|\zeta_2\|_2} & \ldots & \frac{\zeta_n}{\|\zeta_n\|_2} \end{bmatrix}$$  \hspace{1cm} (3.4)

An example of the RVD aggregation scheme is shown in Figure 3.8.
3.2 Intra stage processing of cluster level representation

We propose modifying the original RVD by increasing the descriptor robustness and discriminative power. The first improvement de-correlates residual vectors $r_{tj}$ by applying cluster level PCA before aggregation. The new representation is named RVD with local PCA (RVD-P) and is presented in subsection 3.2.1. The second method aims to balance the variances of different dimensions of residual vectors $r_{tj}$ after PCA transformation. It is called RVD with local Whitening (RVD-W) and described in subsection 3.2.2.

3.2.1 RVD with local PCA (RVD-P)

We improved the performance of RVD signature by transforming the weighted residual vectors $r_{tj} = \frac{x - \mu_j}{||x - \mu_j||_1}$ inside each cluster using a local PCA basis $P_j$ before aggregation into RVD-P. In the following, we describe the process of computing RVD-P representation.

**Off-line stage:** Given a set of $N$ weighted residual vectors $r_{1j}, r_{2j}, \ldots, r_{Nj}$ in $\mathbb{R}^d$ extracted from $I$ training images, we compute the mean vector $\eta_j = \mathbb{E}[r_{tj}]$ and the
3.2. Intra stage processing of cluster level representation

Figure 3.8: RVD aggregation approach: The solid polygons indicate Voronoi cells. In Figure 3.8 (a) there are six local descriptors $x_1, \ldots, x_6$ and five cluster centres $\mu_1, \ldots, \mu_5$. The descriptor $x_1$ is assigned to its three nearest clusters centres ($\mu_1, \mu_2, \mu_3$), and the corresponding residual vectors, $(x_1 - \mu_1)$, $(x_1 - \mu_2)$, $(x_1 - \mu_3)$, are respectively shown as red, green and orange arrows. The descriptors $\{x_1, x_4, x_6\}$ are assigned to their first nearest cluster ($\mu_1$) and the residual vectors $(x_t - \mu_1)$ are L1-normalized (shown by scaling the red coloured residual vectors to dashed unit square) in order to discard the magnitude information. The normalized residual vectors are aggregated into $rv_1$ as shown in Figure 3.8 (b). Similarly, descriptors $\{x_2, x_3, x_5\}$ are quantized to their second nearest cluster $\mu_1$ and the L1-normalized residual vectors (shown by green arrows) are aggregated into $rv_2$ (Figure 3.8(c)). Finally, $rv_1$ and $rv_2$ are combined with rank assignment weights $\tau_t$ into RVD cluster level representation $\zeta_1$ (shown in Figure 3.8(d)). The figure only shows two ranks for simplicity.
covariance matrix $\Sigma_j$ for each cluster $j$:

$$\eta_j = \frac{1}{N_j} \sum_{i=1}^{I} \sum_{\gamma=1}^{K} \sum_{x_t:NN^*_\gamma(x_t)=j} r_{tj}$$ (3.5)

$$\Sigma_j = \frac{1}{N_j} \sum_{i=1}^{I} \sum_{\gamma=1}^{K} \sum_{x_t:NN^*_\gamma(x_t)=j} (r_{tj} - \eta_j)(r_{tj} - \eta_j)^\top$$ (3.6)

For each cluster $j$, we compute a PCA matrix $P_j$ whose columns consists of the orthonormal eigenvectors of $\Sigma_j$ corresponding to the $d$ largest eigenvalues $\lambda_1 \geq \lambda_2 \ldots \geq \lambda_d$.

**On-line Stage:** Given an image, the vectors $r_{tj}$ are extracted for each cluster $j$ as in the core RVD method. The mean subtracted vectors $r_{tj}$ are projected using $P_j$ before aggregation into cluster level representation $\zeta_j$:

$$\zeta_j = \sum_{\gamma=1}^{K} \sum_{x_t:NN^*_\gamma(x_t)=j} P_j^\top (r_{tj} - \eta_j)$$ (3.7)

The final RVD-P representation $R^p$ is formed by concatenating L2-normalized $\zeta_j$ vectors for all clusters:

$$R^p = \left[ \frac{\zeta_1}{\|\zeta_1\|_2} ; \frac{\zeta_2}{\|\zeta_2\|_2} ; \ldots ; \frac{\zeta_n}{\|\zeta_n\|_2} \right]$$ (3.8)

**Comparison of cluster level PCA and LDA**

In this section we study the impact of projecting residuals vectors using two different transformations: (1) cluster-level PCA and (2) cluster-level linear discriminant analysis (LDA) [23].

It can be observed from Figure 3.9 that transforming residual vectors $r_{tj}$ using cluster-specific PCA leads to a more robust and discriminative RVD signature thus providing better retrieval accuracy on both Holidays and Oxford datasets.

**3.2.2 RVD with local Whitening (RVD-W)**

Figure 3.10 shows the energy distribution in each dimension of residual vectors $r_{tj}$ before aggregation into RVD (blue line) and it can be observed that the variances of
3.2. Intra stage processing of cluster level representation

Figure 3.9: Comparison of cluster level PCA and LDA (a) Holidays, (b) Oxford5k

Different dimensions are not balanced, which negatively affects the discriminability of the final global representation. We solve the aforementioned problem by introducing whitening of the residual vectors $r_{tj}$ before aggregation into cluster level representation.

More precisely, we compute the cluster level whitening matrix $P^w_j$ as $P^w_j = P_j \Lambda_j^{-\frac{1}{2}}$, where $\Lambda_j = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d)$. Given an image, the vectors $r_{tj}$ are computed for each cluster $j$. The mean subtracted $r_{tj}$ vectors are then projected using $P_j$ and subsequently whitened before aggregation into $\zeta_j$:

$$
\zeta_j = \sum_{\gamma=1}^{K} \sum_{x_t \in N^K_{j}(x_t)=j} P^w_j (r_{tj} - \eta_j) \quad (3.9)
$$

The L2-normalized $\zeta_j$ vectors are stacked to form the final RVD-W representation $R^w$:

$$
R^w = \left[ \frac{\zeta_1}{\|\zeta_1\|_2}, \frac{\zeta_2}{\|\zeta_2\|_2}, \ldots, \frac{\zeta_n}{\|\zeta_n\|_2} \right] \quad (3.10)
$$

It can be observed from Figure 3.10 that in RVD-P aggregation, the application of local PCA on $r_{tj}$ concentrates the energy in top few dimensions while in RVD-W, after performing PCA+Whitening the energy remains balanced between dimensions. Figure 3.11 presents the performance comparison between RVD, RVD-P and RVD-W representations. It can be clearly seen that RVD-W on average provides the best performance on both Holidays and Oxford5k datasets.
3.2. Intra stage processing of cluster level representation

Figure 3.10: The energy distribution in each dimension of residual vectors $r_{ij}$ before aggregation into RVD, RVD-P and RVD-W respectively

![Graph showing energy distribution in each dimension of residual vectors](image)

Figure 3.11: RVD-W comparison with RVD-P and RVD (a) Holidays, (b) Oxford5k

![Graph showing mAP (%) vs Global Descriptor Dimension for Holidays and Oxford5k](image)

3.2.3 PCA transformation and L1+Power normalization

In order to improve the separability between matching and non-matching representations, we propose a new normalization approach applied after transforming the RVD-W vectors via PCA. Our normalization involves an L1-norm followed by a power-norm creating L1-P normalization. We show that the L1-P is different to the frequently used Whitening [48] and Power+L2 normalization and offers significant gains in terms of
3.2. Intra stage processing of cluster level representation

Figure 3.12: Histogram of Euclidean similarity between matching and non-matching descriptors, for three post-PCA normalization methods (a) Whitening (b) P-L2 (c) L1-P.

retrieval accuracy.

1) **Whitening**: In [48], Jegou et al. applied whitening operation on the VLAD vector in order to increase the contrast between matching and non-matching descriptors. We follow [48] and perform whitening on RVD-W vector to evaluate its impact on retrieval performance. More precisely, the mean-centred $R_w$ vector is first PCA-transformed, and subsequently whitened and re-normalized to form vector $R_{whit}$:

$$R_{whit} = \frac{\text{diag}(\lambda_{1}^{-1/2}, \ldots, \lambda_{D'}^{-1/2})P(R_w - R_0)}{||\text{diag}(\lambda_{1}^{-1/2}, \ldots, \lambda_{D'}^{-1/2})P(R_w - R_0)||_2}$$  \hspace{1cm} (3.11)

where $R_0$ is the mean of the signatures of $R_w$ and $P$ is a $D' \times D$ matrix $(D' \leq D)$ of eigenvectors associated with the largest eigenvalues of the covariance matrix of signatures of $R_w$.

2) **Power+L2 normalization (P-L2)**: The whitening of RVD-W vectors is only suitable when generating short signatures because the smallest eigenvalues produce artefacts. Figure 3.13(c) demonstrates that the retrieval accuracy initially increases up to 512 dimensions but then decreases when the dimensionality of the RVD-W vector exceeds 512. [47] addressed this problem by applying power-normalization on the PCA projected descriptor, followed by L2-normalization. The power-norm is parametrized by a constant $\beta$.

3) **L1+Power normalization (L1-P)**: In our approach, the mean-centred $R_w$ vector
is first transformed using matrix $P$ and then the resultant vector is L1-normalized to form $R^{\text{wo}}$:

$$R^{\text{wo}} = \frac{P(R^w - R_0)}{||P(R^w - R_0)||_1}$$  \hspace{1cm} (3.12)

Finally, the vector $R^{\text{wo}} = (R^{\text{wo}}_1, ..., R^{\text{wo}}_{D'})$ is processed using power-normalization: $R^{\text{wl}}_i = \text{sign}(R^{\text{wo}}_i)|R^{\text{wo}}_i|^{\beta}$.

We use the class-separability between matching and non-matching descriptors to demonstrate the advantage of our approach, on MPEG dataset (10k matching and 100k non-matching image pairs) [40]. More precisely, the dimensionality of $R^w$ is reduced to 512 and post-PCA normalization is applied. Let us denote $Pr(h|m)$ and $Pr(h|nm)$ as the probability density function (pdf) of observing a Euclidean distance $h$ for a matching and non-matching descriptor pair respectively. The distance between matching/non-matching pdfs is expressed in terms of KL-divergence. It can be observed from Figure 3.12 that L1-P method provides the best separability (maximum KL-Divergence) between matching and non-matching distributions, compared to Whitening and P-L2 approaches.

We study how the power-normalization exponent $\beta$ of P-L2 and L1-P normalizations, effects the retrieval performance of RVD-W and FV. From Figure 3.13(a) and Figure 3.13(b), it can be observed that L1-P normalization ($\beta = 0.7$), provides close to optimum performance for both large dimensional (D’=8192) and small dimensional (D’=128) RVD-W descriptor. It is interesting to note that similar behaviour is also shown by the FV representation. Experiments are conducted to compare the performance of the three post PCA normalization methods: (i) Whitening, (ii) P-L2 normalization ($\beta = 0.5$), and (iii) L1-P normalization ($\beta = 0.7$). It can be clearly seen from Figure 3.13(c) and Figure 3.13(d) that normalizing the PCA-projected vector using L1-P normalization provides significantly better retrieval accuracy on both the Holidays and the Holidays1M datasets.
3.3 Compact global descriptor

The descriptor size expressed by bytes per image, has a major impact on the performance of an image retrieval system; ideally the descriptors for the entire dataset should fit in the RAM memory of the server for fast processing. Aggregating a 128-dimensional local descriptor (e.g. SIFT) using a small visual vocabulary of 64 visual words results in 8k-dimensional global descriptor. This size is too large for efficient retrieval in very large databases. Recently several notable algorithms have been introduced to compress real-valued global image descriptors to compact codes. In [53], an effective vector quantization method called Product Quantization (PQ) is proposed. In this approach the
3.4 Experimental Setup and Evaluation

The purpose of this section is to evaluate the compact version of RVD-W relative to other state-of-the-art global image representations. We first present the experimental setup which includes the databases and evaluation protocols. Furthermore, we also define common conditions concerning local descriptor extraction, dimensionality reduction of local descriptors and selection of vocabulary size. A comparison with the global descriptor is first projected using a $D \times D'$ PCA matrix and then the truncated vector is divided into $m$ sub-vectors or groups of equal length $D'/m$. Each sub-vector is quantized using a separate K-means quantizer with $n$ centroids (typically 256) and encoded using $k = \log_2(n)$ bits. The storage requirement of the embedded vector is $B = m \times k$ bits.

Since the PQ is applied after PCA projection, the variance in each dimension of the embedded vector is not balanced: the first sub-vectors have higher variance than the last. This results in suboptimal performance, since PQ allocates a fixed number of bits to each sub-vector. The solution is to balance the variance between the sub-vectors so that all the quantizers encode sub-vectors with similar variance. In [54], Jegou et al. applied an orthogonal transformation matrix on PCA projected vectors, prior to PQ encoding. However the drawback of this approach is that the individual product quantizers lose independence (i.e. the de-correlation effect of PCA is lost). In [33], Tiezheng et al. introduced the Eigenvalue Allocation algorithm in which the dimensions of the PCA projected vector are permuted before quantization.

We followed [33] to compress RVD-W vectors into small codes for large scale retrieval. More precisely, we learn a matrix $P$ whose columns correspond to eigenvectors associated with the $D'$ largest eigenvalues of the covariance matrix of signature $R_w$ and use the Eigenvalue Allocation method to re-order the columns of $P$ to form matrix $P'$. The mean subtracted $R_w$ vector is first projected by $P'$, LP normalized and finally the PQ algorithm is applied to the normalized vector. The distance between query vector and database vectors is computed using Asymmetric Distance Computation (ADC) method [54].
state-of-the-art global representations namely FV, VLAD, Temb, RVD, RVD-P and RVD-W, is presented at the end of this section.

### 3.4.1 Datasets

The retrieval accuracy of the our method is extensively evaluated on three standard image retrieval benchmarks: INRIA Holidays, the University of Kentucky Recognition Benchmark (UKB) and the Oxford building dataset. Independent datasets are used for all learning stages, to limit any potential bias introduced by model over-fitting.

The **INRIA Holidays** dataset [51] comprises of 1491 holiday photos with a subset of 500 used as queries. The retrieval accuracy is evaluated using mean Average Precision (mAP), as defined in [90]. To evaluate system performance in a more challenging retrieval scenario, the Holidays dataset is augmented with 1 million distractor images obtained from Flickr, forming Holidays1M [51]. We also further extend Holidays1M with additional 9M distractor images (ImageNET fall 2011 release URLs) [29] to test the robustness of our framework in a very large scale case. The PCA transformation matrix and visual vocabulary is trained on the Flickr60K dataset [51].

The **University of Kentucky Benchmark** (UKB) [82] dataset comprises of 10200 images of 2550 objects. The performance measure is the average number of images returned in the first 4 positions ($4 \times \text{Recall@4}$).

The **Oxford5k** dataset [90] contains 5062 images gathered from Flickr by querying for particular Oxford landmarks. From this set of images, 11 distinctive landmarks are selected, with 5 distinct queries per landmark. The performance is evaluated using mAP. To test large scale retrieval, this dataset is combined with 100k and 1 million Flickr images [17], forming the Oxford105k [90] and Oxford1M dataset respectively. The Oxford1M dataset is also augmented with 9M distractor images [29] forming Oxford10M dataset. We have used the Paris6k dataset [91] for learning of parameters (PCA and vocabulary).
3.4. Experimental Setup and Evaluation

3.4.2 Local descriptor extraction

In all our experiments, keypoints are detected using the Hessian affine detector [75] and local regions are encoded in a 128-dimensional SIFT descriptor [70]. We use the publicly available SIFT descriptors [51] for Holidays and Holidays1M datasets; while for Oxford datasets, the detector and the SIFT descriptors are computed as in [8]. Descriptors are extracted from the UKB and ImageNET datasets (Holidays10M) using software available on-line [54]. The SIFT descriptors are converted to RootSIFT [7] without any additional storage or memory.

3.4.3 Dimensionality reduction on local descriptors

We formed a hypothesis, based on the research papers [54] [88], that the dimensionality reduction of SIFT features via PCA is essential in-order to obtain good retrieval performance for RVD-based representations because

- PCA de-correlates the data, which is beneficial to both RVD and RVD-W representations.

- Dimensionality reduction removes the less energetic components thereby improving the discriminatory power of RVD and RVD-W signatures.
To confirm that these hypotheses are valid, we compute the performance of the RVD and the RVD-W representation on the Holidays dataset as a function of SIFT dimensionality $d$. It can be observed from Figure 3.14(a) that applying PCA and truncating the last 64 dimensions provides the optimum performance.

### 3.4.4 Vocabulary size

In this experiment, the impact of vocabulary size on the retrieval performance of RVD and RVD-W was studied. It can be observed from Figure 3.14(b) that the performance increases as we increase the number of centroids. For $n = 256$ RVD-W obtains a mAP=77.2% on the Holidays dataset. However there are two drawbacks of using higher values of $n$: (1) the dimensionality of the global descriptor becomes prohibitive for large scale experiments (2) Higher dimensional global descriptors suffer more when dimensionality reduced via PCA. Table 3.1 clearly shows that the retrieval accuracy (mAP) deteriorates more when RVD-W is truncated from 16k to 128 dimensions compared to RVD-W (8k).

<table>
<thead>
<tr>
<th>Descriptor</th>
<th>$n$</th>
<th>$D$</th>
<th>$D' = D$</th>
<th>$D' = 512$</th>
<th>$D' = 128$</th>
</tr>
</thead>
<tbody>
<tr>
<td>RVD-W</td>
<td>128</td>
<td>8k</td>
<td>76.5</td>
<td>72.1</td>
<td>66.9</td>
</tr>
<tr>
<td>RVD-W</td>
<td>256</td>
<td>16k</td>
<td>77.2</td>
<td>71.2</td>
<td>65.8</td>
</tr>
</tbody>
</table>

Table 3.1: Effect of dimensionality reduction on RVD-W descriptors for $n = 128$ and $n = 256$ (Holidays dataset).

In all the following experiments, the size of codebook is fixed at 128 to provide a good trade-off between performance, extraction speed and memory use.

### 3.4.5 Comparison of RVD/RVD-P/RVD-W/FV/Temb

In this section we compare the best representation RVD-W with RVD, RVD-P, FV and Temb ($\phi_{\Delta} + \psi_{d}$). It can be clearly seen from Figure 3.15 that RVD-W on average outperforms all global descriptors. Compared to FV, RVD-W offers an average gain of +4.5% and +3% in mAP on the Holidays and Oxford5k datasets. The average
3.4. Experimental Setup and Evaluation

Figure 3.15: RVD-W comparison with RVD-P, RVD, FV and Temb ($\phi_\Delta + \psi_d$) (a) Holidays, (b) Oxford5k, (c) UKB, (d) Oxford105k, (e) Holidays1M and (f) Oxford1M (all results in mAP(%) except for recall@4 for UKB);
difference in retrieval performance is even more significant on large scale datasets of Holidays1M (+7%) and Oxford1M (+3.9%) compared to FV. We also compared RVD-W with the recent $\phi_\Delta + \psi_d$ representation. It can be observed that $\phi_\Delta + \psi_d$ obtains marginally better mAP than RVD-W on Holidays and Oxford datasets using a 8192 dimensional descriptor. However $\phi_\Delta + \psi_d$ descriptor suffers significantly from dimensionality reduction and also the computation of $\phi_\Delta + \psi_d$ is typically three orders of magnitude slower than RVD-W. The retrieval performance of RVD-W is significantly better than $\phi_\Delta + \psi_d$ (+5.4% and +6% on Holidays and Oxford5k) after the global descriptors are dimensionally reduced to $D'=128$. On large scale dataset of Holidays1M, RVD-W offers a significant gain of +6% in mAP as compared to $\phi_\Delta + \psi_d$.

### 3.4.6 Optimized Product Quantization

The purpose of this section is to evaluate the RVD-W representation when employed with the joint dimensionality reduction and OPQ method of Section 3.3. The dimensionality of global descriptor is reduced from 8192 to 128 using matrix $P'$. The truncated descriptor is LP normalized and finally the PQ algorithm is applied on the normalized vector. In all the experiments we used $m=16$ sub-vectors and 8 bits to encode each sub-vector resulting in a small code of 16 bytes. The similarity between query descriptor and the database descriptor is computed using ADC.

Table 3.2 shows the performance of compact RVD, RVD-P, RVD-W and Fisher Vector. It can be seen that RVD-W consistently shows better performance on all datasets, achieving 5% higher mAP on the Holidays1M and the Oxford1M compared to FV.

### 3.4.7 Large scale experiments

Figure 3.16(a) and Figure 3.16(b) display the performance of our method on the large scale datasets of Holidays10M and Oxford10M. The mAP performance is presented as a function of dataset size. We show the results for four cases:

- the RVD-W vector is reduced to $D'=128$ dimensions by PCA
Table 3.2: RVD, RVD-P, RVD-W and FV performance using 16 bytes codes. All results are presented in terms of mAP(%).

<table>
<thead>
<tr>
<th>Method</th>
<th>Holidays</th>
<th>Oxford5k</th>
<th>Hol1M</th>
<th>Oxf1M</th>
</tr>
</thead>
<tbody>
<tr>
<td>FV</td>
<td>56.3</td>
<td>38.1</td>
<td>31.0</td>
<td>24.7</td>
</tr>
<tr>
<td>RVD</td>
<td>58.1</td>
<td>39.0</td>
<td>33.4</td>
<td>26.8</td>
</tr>
<tr>
<td>RVD-P</td>
<td>59.2</td>
<td>39.7</td>
<td>36.3</td>
<td>27.5</td>
</tr>
<tr>
<td>RVD-W</td>
<td><strong>61.4</strong></td>
<td><strong>41.2</strong></td>
<td><strong>37.3</strong></td>
<td><strong>29.0</strong></td>
</tr>
</tbody>
</table>

- the Fisher vector is reduced to $D'=128$ by PCA
- the RVD-W vector is first projected by $P'$ matrix and then encoded to 16 bytes using the $16 \times 8$ PQ scheme.
- the Fisher vector compressed to 16 bytes using the $16 \times 8$ PQ scheme.

The retrieval performance demonstrates that the RVD-W representation consistently and significantly outperforms FV for both Oxford10M and Holiday10M datasets, typically by a margin of 6% in mAP. Interestingly, it can also be observed that the performance gap increases as dataset size grows, particularly for the more difficult Oxford dataset. This indicates that RVD-W is more robust in large-scale retrieval. On ultra large scale dataset of Holidays10M, RVD-W ($D'=128$) obtains a mAP of 40.5% which significantly outperforms any results published to date. To the best of our knowledge, this is the first time that the retrieval experiments are performed on the Oxford dataset enlarged to 10M.

In order to evaluate the performance of our global descriptor in a retrieval system where a short list of images retrieved by RVD-W is re-ranked using local descriptor matching with geometric verification, we evaluated Recall@L i.e. the number of relevant images retrieved in the top L returns. The results are shown in Figures 3.16(c) and 3.16(d) where it can be seen that the RVD-W representation is significantly better than FV in returning correct matches.

To illustrate the retrieval performance, we compress RVD-W and FV vectors of the Oxford1M dataset, using OPQ, to obtain small codes of 16 bytes. The distance be-
Figure 3.16: Retrieval performance as a function of the database size (a) Holidays10M and (b) Oxford10M. Quality of short-list: recall@L (c) Holidays1M and (d) Oxford1M.

between a query vector and database vectors is computed using ADC and for every query Recall@100 is calculated. We observe that, out of a total of 55 queries, RVD-W obtains better recall on 20 queries and FV has better recall on 7 queries; the ratio of RVD-W outperforming FV is approximately 3:1. For an intuitive understanding, Fig. 3.17 shows three queries where the difference in recall between RVD-W and FV is most significant and one query where the difference in recall between FV and RVD-W is the biggest (maintaining the 3:1 ratio established before). We show the query and the top 4 ranked results obtained by the RVD-W and FV methods using these queries, correct matches are indicated by a blue frame.
Figure 3.17: Example retrieval results for the RVD-W and FV descriptors on the Oxford1M dataset. For each Query image (left) the corresponding ranked lists are shown for the RVD-W (top centre-right) and FV (bottom centre-right); images correctly retrieved are marked with blue border. Both descriptors are quantized using OPQ to 16 bytes.
3.5 Comparison with the state of the art

In this section we compare the performance of the proposed method to the latest state-of-the-art algorithms using full-scale and compact representations. Table 3.3 summarises the results for uncompressed, highly-dimensional representations. In practical applications, the use of full-dimensional vectors is prohibitive due to search time and memory requirements, however the results are helpful in understanding the capabilities of each representation, and also serve as an upper bound on the expected performance of compact descriptors derived from them. It can be seen that the proposed RVD-W representation outperforms all prior-art methods, in particular it improves dramatically (gain of +10% mAP) over the most advanced version of VLAD [28] (referred here as VLAD_{LCS+RN}) and also over FV (gain of +16%), on both Holidays and Oxford databases. Compared to the latest method based on triangulation embedding with sum aggregation ($\phi_\Delta + \psi_d$) [47], RVD-W ($D = 8192$) provides a significant improvement of +3.5%, +2% and +8.5% in mAP on the Oxford, Holidays and Oxford105k datasets. The $\phi_\Delta + \psi_d$ representation performs marginally better than RVD-W on Holidays and Oxford datasets using a 8192 dimensional descriptor. However $\phi_\Delta + \psi_d$ descriptor suffers significantly from dimensionality reduction as shown by the sharp decrease in performance when the $\phi_\Delta + \psi_d$ descriptor is truncated from 8192 to 1024 dimensions, compared to RVD-W. Also the extraction time of $\phi_\Delta + \psi_d$ is typically three orders of magnitude slower than RVD-W. On large scale dataset of Oxford105k, RVD-W offers a gain of +2.9% compared to $\phi_\Delta + \psi_d$. By increasing the number of cluster to 256 the RVD-W (16k) outperforms the $\phi_\Delta + \psi_d$ signature on all datasets.

We now focus on comparison of compact representations which are practicable in large-scale retrieval, as presented in Table 3.4. The dimensionality of RVD-W descriptor is reduced from 8192 to 128 via PCA. The results show that our method outperforms all presented methods by a large margin. The gain over Fisher vector is +16% and +10% respectively for Oxford and Holidays datasets. The RVD-W provides an improvement of 13.9% and 16% on the Oxford5k and Oxford105k datasets over VLAD_{LCS+RN}. Keeping the original descriptor dimensionality of 8192, RVD-W offers gains of +6% and +5% in mAP on the Oxford and Holiday datasets compared to the $\phi_\Delta + \psi_d$. It should be
noted that no results are published for 8192 dimensional \( \phi + \psi_d \) on Holidays1M dataset because of extremely high encoding times. Compared to \( \phi + \psi_d \) (D=1920), our method provides an improvement of 6.4% on Holidays1M. It is worth mentioning that \( \phi + \psi_d \) (D=1920) suffers less from dimensionality reduction compared to \( \phi + \psi_d \) (D=8192).

On an ultra large dataset of Holidays10M, RVD-W significantly outperforms the best published results (VLAD+SURF).

Table 3.5 shows the performance of compact RVD-W obtained by OPQ algorithm. Compared to VLAD\_LCS+RN, the advantage remains very significant on Oxford5k (+14%), Oxford105k (+16%) and Holidays1M (+5%). The RVD-W provides a gain of 9.4% on largest Holidays10M dataset over VLAD+SURF.

In summary, RVD-W has proven to be a very robust and high performing representation with its compressed versions delivering world-class performance in large-scale retrieval.

### 3.6 Summary

This chapter presents a novel method for extraction of a robust and highly discriminative global descriptor. The key ideas include a novel robust aggregation approach with rank-based multi-assignment, direction-based accumulation, and mid-stage decorrelation and whitening of the residual vectors. A new post-processing method involving a L1 norm and power normalization is also proposed, securing further significant performance gains. We also show that this post-processing method is of general benefit to other existing methods, such as VLAD or FV. A detailed evaluation on de-facto standard benchmarks demonstrates that our scheme outperforms all published state-of-the-art methods.
3.6. Summary

Table 3.3: Comparison with the-state-of-the-art using full dimensional vectors on Oxford5k, Oxford105k, Holidays and UKB datasets. The representation 8k→1k denotes that the global descriptor dimensionality is reduced from 8192 to 1024 via PCA. All results are presented in terms of mAP(%) except for recall@4 for UKB.

<table>
<thead>
<tr>
<th>Method</th>
<th>Size</th>
<th>Oxford5k</th>
<th>Oxford105k</th>
<th>Holidays</th>
<th>UKB</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoW [54]</td>
<td>20k</td>
<td>35.4</td>
<td>-</td>
<td>43.7</td>
<td>2.87</td>
</tr>
<tr>
<td>BoW [54]</td>
<td>200k</td>
<td>36.4</td>
<td>-</td>
<td>54.0</td>
<td>2.81</td>
</tr>
<tr>
<td>VLAD [54]</td>
<td>4k</td>
<td>37.8</td>
<td>-</td>
<td>55.6</td>
<td>3.28</td>
</tr>
<tr>
<td>VLAD [54]</td>
<td>16k</td>
<td>58.7</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FV [54]</td>
<td>4k</td>
<td>41.8</td>
<td>-</td>
<td>60.7</td>
<td>3.35</td>
</tr>
<tr>
<td>VLAD Intra [8]</td>
<td>32k</td>
<td>55.8</td>
<td>-</td>
<td>65.3</td>
<td>-</td>
</tr>
<tr>
<td>VLAD* [28]</td>
<td>8k</td>
<td>50.0</td>
<td>44.5</td>
<td>62.2</td>
<td>-</td>
</tr>
<tr>
<td>VLAD\textsubscript{LCS+RN} [28]</td>
<td>8k</td>
<td>51.7</td>
<td>45.6</td>
<td>65.8</td>
<td>-</td>
</tr>
<tr>
<td>PVLAT [80]</td>
<td>250k</td>
<td>54.2</td>
<td>-</td>
<td>66.4</td>
<td>-</td>
</tr>
<tr>
<td>CPVLAT [80]</td>
<td>9k</td>
<td>-</td>
<td>-</td>
<td>70.0</td>
<td>-</td>
</tr>
<tr>
<td>HVLAD [31]</td>
<td>8k</td>
<td>47.2</td>
<td>-</td>
<td>69.1</td>
<td>-</td>
</tr>
<tr>
<td>VLAD+SURF [120]</td>
<td>4k</td>
<td>32.8</td>
<td>-</td>
<td>64.9</td>
<td>3.20</td>
</tr>
<tr>
<td>VLAD+CSURF [120]</td>
<td>12k</td>
<td>-</td>
<td>-</td>
<td>71.7</td>
<td>3.52</td>
</tr>
<tr>
<td>HiVLAD [69]</td>
<td>8k</td>
<td>57.6</td>
<td>-</td>
<td>66.6</td>
<td>3.48</td>
</tr>
<tr>
<td>HiVLAD [69]</td>
<td>32k</td>
<td>63.8</td>
<td>-</td>
<td>72.1</td>
<td>3.56</td>
</tr>
<tr>
<td>φ\textsubscript{Δ} + ψ\textsubscript{d} [47]</td>
<td>8k</td>
<td>67.6</td>
<td>61.1</td>
<td>77.1</td>
<td>-</td>
</tr>
<tr>
<td>φ\textsubscript{Δ} + ψ\textsubscript{d} [47]</td>
<td>8k→1k</td>
<td>56.2</td>
<td>50.2</td>
<td>72.0</td>
<td>-</td>
</tr>
<tr>
<td>φ\textsubscript{Δ} + ψ\textsubscript{s} [47]</td>
<td>8k</td>
<td>63.3</td>
<td>55.5</td>
<td>74.5</td>
<td>-</td>
</tr>
<tr>
<td>RVD</td>
<td>8k</td>
<td>63.8</td>
<td>61.0</td>
<td>73.5</td>
<td>3.53</td>
</tr>
<tr>
<td>RVD-P</td>
<td>8k</td>
<td>65.1</td>
<td>62.3</td>
<td>74.8</td>
<td>3.55</td>
</tr>
<tr>
<td>RVD-W</td>
<td>8k</td>
<td>66.8</td>
<td>64.0</td>
<td>76.5</td>
<td>3.59</td>
</tr>
<tr>
<td>RVD-W</td>
<td>8k→1k</td>
<td>59.0</td>
<td>56.1</td>
<td>73.2</td>
<td>3.56</td>
</tr>
<tr>
<td>RVD-W</td>
<td>16k</td>
<td><strong>68.9</strong></td>
<td><strong>66.0</strong></td>
<td><strong>77.2</strong></td>
<td><strong>3.6</strong></td>
</tr>
</tbody>
</table>
Table 3.4: Comparison with the state-of-the-art using 96/128 dimensional vectors on Oxford5k, Oxford105k, Holidays, Holidays1M and Holidays10M datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>Size</th>
<th>Oxford 5k</th>
<th>Oxford 105k</th>
<th>Holidays 1M</th>
<th>Holidays 10M</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAD [54]</td>
<td>128</td>
<td>28.7</td>
<td>-</td>
<td>55.7</td>
<td>-</td>
</tr>
<tr>
<td>FV [54]</td>
<td>96</td>
<td>-</td>
<td>-</td>
<td>56.0</td>
<td>31.8</td>
</tr>
<tr>
<td>FV [54]</td>
<td>128</td>
<td>30.1</td>
<td>-</td>
<td>56.5</td>
<td>-</td>
</tr>
<tr>
<td>VLAD* [28]</td>
<td>128</td>
<td>32.5</td>
<td>26.6</td>
<td>-</td>
<td>33.5</td>
</tr>
<tr>
<td>VLAD_{LCS+RN} [28]</td>
<td>128</td>
<td>32.2</td>
<td>26.2</td>
<td>-</td>
<td>39.2</td>
</tr>
<tr>
<td>CPVLAT [80]</td>
<td>256</td>
<td>-</td>
<td>-</td>
<td>60.6</td>
<td>38.0</td>
</tr>
<tr>
<td>VLAD+SURF [120]</td>
<td>96</td>
<td>-</td>
<td>-</td>
<td>65.5</td>
<td>42.5</td>
</tr>
<tr>
<td>HiVLAD [69]</td>
<td>128</td>
<td>-</td>
<td>-</td>
<td>64.0</td>
<td>43.0</td>
</tr>
<tr>
<td>$\phi_\Delta + \psi_d$ [47]</td>
<td>8k→128</td>
<td>40.0</td>
<td>33.9</td>
<td>61.5</td>
<td>-</td>
</tr>
<tr>
<td>$\phi_\Delta + \psi_d$ [47]</td>
<td>2k→128</td>
<td>43.3</td>
<td>35.3</td>
<td>61.7</td>
<td>38.7</td>
</tr>
<tr>
<td>RVD</td>
<td>128</td>
<td>44.5</td>
<td>40.5</td>
<td>63.5</td>
<td>41.9</td>
</tr>
<tr>
<td>RVD-P</td>
<td>128</td>
<td>45.1</td>
<td>41.5</td>
<td>64.0</td>
<td>44.1</td>
</tr>
<tr>
<td>RVD-W</td>
<td>128</td>
<td>46.1</td>
<td>42.5</td>
<td>66.9</td>
<td>45.1</td>
</tr>
</tbody>
</table>

Table 3.5: Comparison with the-state-of-the-art with compact codes via PQ on Oxford5k, Oxford105k, Holidays, Holidays1M and Holidays10M datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>Size</th>
<th>Oxford 5k</th>
<th>Oxford 105k</th>
<th>Holidays 1M</th>
<th>Holidays 10M</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAD [54]</td>
<td>40 B</td>
<td>-</td>
<td>-</td>
<td>49.5</td>
<td>-</td>
</tr>
<tr>
<td>FV [74]</td>
<td>16 B</td>
<td>-</td>
<td>-</td>
<td>50.6</td>
<td>28.7</td>
</tr>
<tr>
<td>VLAD* [28]</td>
<td>16 B</td>
<td>28.9</td>
<td>22.2</td>
<td>-</td>
<td>29.9</td>
</tr>
<tr>
<td>VLAD_{LCS+RN} [28]</td>
<td>16 B</td>
<td>27.0</td>
<td>21.0</td>
<td>-</td>
<td>32.3</td>
</tr>
<tr>
<td>VLAD+SURF [120]</td>
<td>10 B</td>
<td>-</td>
<td>-</td>
<td>58.0</td>
<td>30.2</td>
</tr>
<tr>
<td>RVD</td>
<td>16 B</td>
<td>39.0</td>
<td>34.7</td>
<td>58.1</td>
<td>33.4</td>
</tr>
<tr>
<td>RVD-P</td>
<td>16 B</td>
<td>39.7</td>
<td>35.4</td>
<td>59.2</td>
<td>36.3</td>
</tr>
<tr>
<td>RVD-W</td>
<td>16 B</td>
<td>41.2</td>
<td>37.1</td>
<td>61.4</td>
<td>37.3</td>
</tr>
</tbody>
</table>
Chapter 4

Binary RVD

Encoding high-dimensional global image representations as compact binary strings provides benefits in storage, extraction and matching speeds, especially for large scale image retrieval tasks. This chapter describes a method for deriving a compact, binary and scalable image descriptor from the core RVD representation presented in Chapter 3. First we show how floating-point and high-dimensional global signatures can be converted to compact binary codes using several prior-art techniques, including PCAE, PCAE+RR and PCAE+ITQ. Importantly, we present a novel approach named PCA Embedding with Weighted Hamming distance, to compute similarity between two binary codes. We perform experiments on image retrieval benchmarks (Holidays, Oxford and UKB) and demonstrate that our method consistently leads to large improvements over state-of-the-art.

4.1 Introduction

As explained in the introduction, todays visual search systems must be highly scalable due to the huge volumes of multimedia data, which can comprise billions of images or video frames. Even with the efficient floating point representations, such as RVD-W, two main challenges for large-scale image search remain:

1. How to derive ultra compact image representations, preferably binary ones, for
4.1. Introduction

fast retrieval directly in the workstation RAM, in order to avoid time consuming
data transfers, and

2. How to develop efficient matching algorithms and strategies that scale up to
hundreds of millions of images.

The first challenge, that of image representation, is a cardinal problem in the area
of computer vision. As explained earlier, the classical retrieval approaches based on
matching local descriptors, do not scale up to image corpora containing millions of im-
ages, because of the high computational complexity of the matching process and large
storage requirements. Chapter 2 presented solutions to aggregate local descriptors to
form global image representations, such as BoW, Fisher Vectors, VLAD, VLAT and
RVD-W. However, these global signatures are still high dimensional, dense and use
floating-point format which makes them unsuitable for large scale retrieval tasks. Re-
cently several notable algorithms have been introduced to compress real-valued global
image descriptors to binary codes. The aim is to transform image representations
from a continuous real-valued space to discrete Hamming space in such a way that the
distances are preserved, and therefore the performance is preserved.

In this chapter we build upon our RVD framework to propose a binary global descrip-
tor suitable for modern challenges of search accuracy, computational efficiency and
low memory use. We present a novel scheme named PCA Embedding with Weighted
Hamming distance (PCAE+WH) for robust and fast matching of global descriptors
which significantly outperforms any results published to date. Our main contributions
include:

- We evaluate the performance of the compressed global descriptors, including
RVD, RVD-W, FV, VLAD, and compare state-of-the-art binary embedding meth-
ods PCAE, PCAE+RR, and PCAE+ITQ. We conclude that the best perfor-
man ce is achieved by PCAE method on all benchmarks and the PCAE+RR and
PCAE+ITQ algorithms do not achieved the expected gain in retrieval accuracy.
The experimental results also show that the RVD and RVD-W signatures outper-
forms FV and VLAD in the binary domain when prior-art binarization techniques
are applied.
• We further investigate PCAE+RR and PCAE+ITQ methods, where the PCA projected global descriptor is transformed via an orthogonal rotation matrix, in order to reduce the quantization error associated with transforming the data to the vertices of the binary Hamming cube. We conclude that applying rotation after PCA significantly deteriorates the retrieval performance due to the fact that the de-correlation achieved by the PCA is effectively lost.

• To solve this aforementioned problem, we propose a novel binary matching method which uses Weighted Hamming distance on the PCAE vectors (PCAE+WH). In our approach the high variance directions are given more weight in the matching stage by suitably weighting the Hamming distance, compared to low-variance directions.

The compression of global descriptors into binary codes is described in Section 4.2. The experimental setup and the detailed evaluation of our method is presented in Section 4.3. In Section 4.4 we compare our results with the state-of-the-art showing significant improvement, for example +4% in mAP on a large scale dataset of Holidays1M or +5% on Oxford1M.

4.2 Compact Global Descriptor

In this section we implement, analyse and evaluate experimentally several successful techniques to convert high-dimensional global descriptors into compact binary signatures. We combine RVD-W representation with PCAE, PCAE+RR and PCAE+ITQ. In the above binarization algorithms the first step is to project the global descriptor $R_w$ using linear or non-linear transformation. The binary signature is then obtained by thresholding the transformed data.

4.2.1 PCA Embedding Binarization (PCAE)

Figure 4.1 shows the pipeline of binarizing RVD-W descriptor using the PCAE method. It involves the following steps:
4.2. Compact Global Descriptor

Figure 4.1: PCA Embedding Binarisation pipeline

1. The local descriptors extracted from an image are aggregated to form RVD-W representation $R^w \in \mathbb{R}^D$ (Eq. 3.9).

2. A PCA transformation matrix $P \in \mathbb{R}^{D' \times D}$ is learned, which contains the eigenvectors corresponding to the $D'$ largest eigenvalues of the covariance matrix of signatures $R^w$.

3. Each global vector $R^w$ is transformed using the learned PCA matrix:

$$R^{wp} = P(R^w - R_0)$$

where $R_0 \in \mathbb{R}^D$ is the mean of the signatures $R^w$.

4. The projected vector $R^{wp} \in \mathbb{R}^{D'}$ is quantized to discrete binary signature $R^{wb}$ using sign-binarisation method:

$$R^{wb} = \left[\text{sign}\{R^{1wp}\}; \text{sign}\{R^{2wp}\}; ...; \text{sign}\{R^{D'wp}\}\right]$$

Due to the application of PCA transformation the dimensions are decorrelated but the energy in different dimensions of $R^{wp}$ is not balanced. Therefore encoding each dimension of the projected vector with equal number of bits leads to significant quantization error on the high energy dimensions. Solution proposed in the prior-art is to balance the energy in each dimension via application of orthogonal rotation matrix, before binarizing the transformed vectors.
4.2. Compact Global Descriptor

Figure 4.2: Energy of different dimensions for PCA (a), RR (b) and ITQ (c) using 1024 dimensional RVD-W representation

4.2.2 PCA Embedding with Random Rotations (PCAE+RR)

In [54], Jegou et al. applied random orthogonal transformation $Q$ to the PCA-projected data in order to balance the energy (variance) of the embedded vectors. The matrix $Q \in \mathbb{R}^{D' \times D'}$ can be generated using any of the following two methods:

1. Generate a random matrix drawn from a standard normal distribution and apply QR decomposition [52].

2. Perform a Singular Value Decomposition (SVD) of the aforementioned random matrix [34].

We follow the second approach to binarize RVD-W using PCAE+RR:

$$R^{\text{eq}} = QP(R^w - R_0)$$  \hspace{1cm} (4.3)

$$R^{\text{wb}} = [\text{sign}\{R^{\text{eq}}_1\}; \text{sign}\{R^{\text{eq}}_2\}; \ldots; \text{sign}\{R^{\text{eq}}_{D'}\}]$$  \hspace{1cm} (4.4)

4.2.3 PCA Embedding with Iterative Quantization (PCAE+ITQ)

In [34], Gong et al. proposed an alternative approach (ITQ) to binarize global vectors, which works by learning an optimal orthogonal transformation $Q \in \mathbb{R}^{D' \times D'}$ to minimize the quantization error $Er$ of transforming the data to the vertices of the binary Hamming cube:

$$Er = ||R^{\text{wb}} - QR^{\text{eq}}||_F^2$$  \hspace{1cm} (4.5)
where $||.||_F$ represents the Frobenius norm.

The ITQ method is related to orthogonal Procrustes problem \[99\], which involves transforming a matrix $A$ into a matrix $B$ by an orthogonal rotation matrix $Q$, such that the quantization error is minimum. More precisely, ITQ is initialized with a orthogonal random rotation $Q$ and an iterative algorithm is adopted to solve the optimization problem. At every iteration the data points are mapped to the nearest vertex of binary Hamming cube and the transformation matrix $Q$ is updated to reduce the quantization error, given this mapping.

The global descriptor $R^w$ is converted to binary code $R^{wb}$ using PCAE+ITQ as:

$$R^{wq} = QP(R^w - R_0)$$

$$R^{wb} = [\text{sign}\{R^{wq}_1\}; \text{sign}\{R^{wq}_2\}; \ldots; \text{sign}\{R^{wq}_{D'}\}]$$

Figure 4.2 shows the energy (variance) of each dimension of RVD-W signatures before and after application of RR and ITQ on PCA-projected data. It can be clearly seen that ITQ balances the energy in each dimension, as expected.

**Analysis and evaluation of binary encoding methods**

We compare the performance of three aforementioned binary encoding methods: PCAE, PCAE+RR and PCAE+ITQ. It can be seen from Figure 4.3 that binarizing RVD-W and FV using PCAE approach provides the best performance on Holidays and Oxford datasets. We explain the low performance achieved by PCAE+RR and PCAE+ITQ methods by the fact that the de-correlation performed by PCA is lost due to application of orthogonal rotation on the projected data. This means that the distances in the feature space can no longer be efficiently represented by the binary strings. Furthermore, the ITQ method performs poorly on high-dimensional data because it directly minimizes the Euclidean distance between real and binary data. However, FV and RVD-W representations utilise cosine similarity, which is a more suitable distance measure than the Euclidean distance.
4.2. Compact Global Descriptor

![Graphs showing mAP (%) vs Global Descriptor Dimension for Holidays and Oxford datasets]

Figure 4.3: Comparison of binary encoding methods. Binary RVDW performance on (a) Holidays, (b) Oxford5k datasets. Binary FV performance on (c) Holidays, (d) Oxford5k datasets.

4.2.4 PCAE with Weighted Hamming distance (PCAE+WH)

A drawback of PCAE is that the variances in each dimension $\ell$ of the PCA-projected vector is not balanced. It can be seen from Figure 4.4 that the variance of dimension 1 of the projected vectors $R^{up}$ is significantly larger than the variance of dimension 256. The ratio in this case is 9. This results in suboptimal performance, as PCAE encodes each dimension using one bit, independently of the energy it carries. In particular, the high variance and low variance dimensions are given equal weight by the Hamming distance operating on the binary signatures. Prior-art proposes to balance the variance in each dimension by applying RR or ITQ algorithms. However as explained previously,
4.2. Compact Global Descriptor

![Graph](image)

Figure 4.4: Distribution of the projected values of 1 million images, (a) projected on the first PCA dimension (b) Projected on the $256^{th}$ PCA dimension

applying rotation after PCA is not effective. We address the aforementioned problem by introducing a novel PCAE+WH approach, where the weights assigned to Hamming distance are proportional to the variance of the projected data in the corresponding dimension. More precisely, each global descriptor $R^w$ is first PCA projected and then binarized using sign function to form vector $R^{wb}$. The similarity score between two binary signatures, $R^{wb}_1$ and $R^{wb}_2$, is computed as a Weighted Hamming distance (WH) between them:

$$Sc = \sum_{\ell=1}^{D'} \Phi(\ell) H(R^{wb}_{1,\ell}, R^{wb}_{2,\ell}) + \Psi(1 - (H(R^{wb}_{1,\ell}, R^{wb}_{2,\ell})))$$  \hspace{1cm} (4.8)

where $\Phi$ and $\Psi$ denotes the weights to the Hamming distance $H$. In the following section, we justify our approach and define the algorithm to compute weights $\Phi$ and $\Psi$.

4.2.5 Computation of Weights $\Phi$ and $\Psi$

In PCAE, a global vector $R^w$ is first projected via a PCA matrix to form vector $R^{wp}$. Due to the application of PCA and the statistical properties of the original descriptors before transformation, the dimensions of $R^{wp}$ are uncorrelated and follow a Gaussian distribution $p_\ell$ with mean $\mu_\ell = 0$ and some standard deviation $\sigma_\ell$. Figure 4.4 shows the
4.2. Compact Global Descriptor

actual distributions of the coefficients of $R^{wρ}$ in dimensions one and 256. Two important
observations can be made: (1) both distributions follow a Gaussian model $p_ℓ$, and (2) the variance along the first dimension is significantly higher. Now, let us assume that the descriptors are binarized using the sign method. In any given dimension $ℓ$, the expectation of distance between points having different signs can be calculated as:

$$E_{diff} = \int_{-\infty}^{0} \int_{0}^{\infty} (u - v)^2 p_ℓ(u)p_ℓ(v) \, du \, dv = \frac{\sigma_ℓ^2(π + 2)}{2π}$$ (4.9)

This value corresponds to $Φ$ in Equation 4.8.

In a particular dimension $ℓ$, the expectation of distance between points having the same sign is given as:

$$E_{same} = \int_{0}^{\infty} \int_{0}^{\infty} (u - v)^2 p_ℓ(u)p_ℓ(v) \, du \, dv = \frac{\sigma_ℓ^2(π - 2)}{2π}$$ (4.10)

This value corresponds to $Ψ$ in Equation 4.8. Therefore, based on Equations 4.9 and 4.10, the weights applied to Hamming distance are proportional to the variance of each dimension of $R^{wρ}$ signatures.

In practice we compute the weights $Φ$ and $Ψ$ experimentally as follows:

1. We randomly draw two sets $V_ℓ = \{v_i, i = 1...I\}$ and $U_ℓ = \{u_i, i = 1...I\}$, from a given dimension $ℓ$ of the $R^{wρ}$ signatures.

2. We divide $V_ℓ$ into two subsets: $V_ℓ^1$ contains the signatures $v_i$ such that $v_i^b = 1$ and $V_ℓ^0$ those signatures $v_i$ where $v_i^b = 0$. Similarly, the set $U_ℓ$ is divided into two subsets $U_ℓ^1$ and $U_ℓ^0$.

3. The weight $Φ_ℓ$ is computed as:

$$Φ_ℓ = \frac{1}{|U_ℓ^0|} \sum_{u \in U_ℓ^0 \land v \in V_ℓ^1} (u - v)^2$$ (4.11)

4. The weight $Ψ_ℓ$ is computed as:

$$Ψ_ℓ = \frac{1}{|U_ℓ^1|} \sum_{u \in U_ℓ^1 \land v \in V_ℓ^1} (u - v)^2$$ (4.12)
To reduce the matching time and memory requirement of PCAE+WH method, we propose computing the Weighted Hamming distance for a group of \( k \) bits rather than weighting each dimension \( \ell \) individually. Such groups could be aligned with the architecture of the system, for example groups of 32 bits (or multiple) can be used for a 32 bit CPU. More precisely, the off-line stage consists of partitioning each set of \( I \) global descriptors into \( m \) groups of equal length \( k = D'/m \) and learning the weights \( \Phi \) and \( \Psi \) for each group. In on-line matching stage (Figure 4.5), a query global descriptor \( R^w_1 \) is first converted to binary code \( R^{wb}_1 \) and then the code is split into \( m \) groups of length \( k \).

The similarity score \( Sc \) between two \( R^{wb} \) vectors is computed as the sum of weighted Hamming distances between the signatures of the corresponding groups:

\[
Sc = \sum_{j=1}^{m} \Phi_j (H(R^{wb}_{1,j}, R^{wb}_{2,j})) + \Psi_j (k - (H(R^{wb}_{1,j}, R^{wb}_{2,j})))
\]  

(4.13)

The score \( Sc \) can be calculated quickly by (i) using bitwise XOR and POPCNT to compute Hamming distances between binary descriptors corresponding to the same groups, and (ii) scaling by the weights stored in a small look-up table.
4.3 Experimental Setup and Evaluation

We first present the experimental setup which includes the databases and evaluation protocols. Next we detail the extraction process of local image descriptors. Following this, we compare the performance of binary global descriptors including VLAD, FV, RVD and RVD-W, using the traditional Hamming distance. Finally, we compare the performance of PCAE coupled with the Hamming distance and the proposed Weighted Hamming distance methods.

4.3.1 Datasets

The evaluation is conducted on three public datasets: INRIA Holidays, the University of Kentucky Recognition Benchmark (UKB) and the Oxford building dataset which were described in subsection 3.4.1. In order to appraise performance for large scale retrieval, a set of 1 million images collected from FLICKR is used (FLICKR1M). The retrieval accuracy on Holidays and Oxford datasets is evaluated in terms of mAP. For UKB dataset, the performance is calculated in terms of $4 \times \text{Recall}@4$.

4.3.2 Local descriptor extraction

Keypoints are computed using the Hessian affine detector [70] and local regions are encoded in a 128-dimensional SIFT descriptor [75]. The SIFT descriptors are projected to 64 dimensional space using PCA.

4.3.3 Comparison of RVD/RVD-P/RVD-W/FV/VLAD

In this section we compare our best performing representation RVD-W with RVD-P, RVD, FV and VLAD. All global representations are converted to compact binary codes using the PCAE method. It can be clearly seen from Figure 4.6 that RVD-W outperforms all global descriptors on all datasets. Compared to FV, RVD-W offers an average gain of $+4\%$ and $+3\%$ in mAP on the Holidays and Oxford5k datasets. The average improvement in retrieval performance is even more compelling on large scale datasets of Holidays1M ($+4.5\%$) and Oxford1M ($+5\%$) over FV.
4.3. Experimental Setup and Evaluation

Figure 4.6: RVD-W comparison with RVD-P, RVD, FV and VLAD (a) Holidays, (b) Oxford5k, (c) UKB, (d) Oxford105k, (e) Holidays1M and (f) Oxford1M (all results in mAP(%) except for recall@4 for UKB);
4.3.4 PCAE with Weighted Hamming distance

This section presents the benefits of the proposed Weighted Hamming distance. We compare the performance of the following two methods:

1. **PCAE + Hamming distance (PCAE+H):** The RVD-W vector $R^w$ (8192 dimensional) is binarized using PCAE to form vector $R^{wb}$ and the standard Hamming distance is employed to compute similarity between two $R^{wb}$ vectors.

2. **PCAE + Weighted Hamming distance (PCAE+WH):** The dimensionality of $R^w$ is reduced from 8192 to $D'$ dimensions and the truncated vector is binarized using zero thresholding to form vector $R^{wb}$. The $R^{wb}$ is then divided into $m$ groups of 32 bits. Given two $R^{wb}$ descriptors, the similarity score $Sc$ is the sum of weighted Hamming distance between the signatures of corresponding groups.

It can be observed from Figure 4.7 that PCAE+WH method significantly reduces the loss in retrieval performance due to quantization of the real-valued descriptors to binary vectors. Compared to the PCAE+H approach, PCAE+WH offers an average gain of +4.5% and +2.2% in mAP on the Holidays and Holidays1M datasets. On UKB benchmark PCAE+WH offers a significant gain of 0.13 in terms of recall@4, compared to PCAE+H method.

For a relatively low number of bits (below 256), the PCAE+H and PCAE+WH methods tend to converge delivering comparable performance on Oxford datasets, with the PCAE+WH gaining clear advantage as the number of bits increases.

We now evaluate the performance of binary global representations RVD-W, RVD-P, RVD and FV, when matching based on weighted Hamming distance is employed. It can be seen that from Figure 4.8 that RVD-W consistently outperforms all global descriptors by a large margin. Compared to FV, RVD-W offers an average gain of +4.5% and +3% in mAP on the Holidays and Oxford5k datasets. On large scale datasets of Holidays1M and Oxford1M, RVD-W provides a dramatic improvement of +5.4% and +4.6% over FV.
4.3. Experimental Setup and Evaluation

Figure 4.7: Comparison of PCAE+H and PCAE+WH methods (a) Holidays, (b) Oxford5k, (c) UKB, (d) Oxford105k, (e) Holidays1M and (f) Oxford1M (all results in mAP(%) except for recall@4 for UKB).
4.3. Experimental Setup and Evaluation

Figure 4.8: Binarization and matching of global descriptors using PCAE+WH (a) Holidays, (b) Oxford5k, (c) UKB, (d) Oxford105k, (e) Holidays1M and (f) Oxford1M (all results in mAP(%) except for recall@4 for UKB);
Table 4.1: Performance of binary global representations on Holidays, Oxford, Holidays1M and UKB datasets, BoW: Bag of Words, FV: Fisher Vectors, VLAD and RVD-W. \textit{FV}_{\text{our}} indicated our implementation of Fisher Vectors. WH denotes Weighted Hamming Distance. All results are presented in terms of mAP(\%) except for recall@4 for UKB.

<table>
<thead>
<tr>
<th>Method</th>
<th>Dim</th>
<th>Distance</th>
<th>Holidays</th>
<th>Oxford</th>
<th>Holidays1M</th>
<th>UKB</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoW [50]</td>
<td>20k</td>
<td>Hamming</td>
<td>45.8</td>
<td>-</td>
<td>-</td>
<td>3.02</td>
</tr>
<tr>
<td>FV [88]</td>
<td>4k</td>
<td>Hamming</td>
<td>57.9</td>
<td>-</td>
<td>-</td>
<td>3.23</td>
</tr>
<tr>
<td>FV [88]</td>
<td>1k</td>
<td>Hamming</td>
<td>52.0</td>
<td>-</td>
<td>-</td>
<td>3.00</td>
</tr>
<tr>
<td>FV [88]</td>
<td>512</td>
<td>Hamming</td>
<td>46.1</td>
<td>-</td>
<td>-</td>
<td>2.80</td>
</tr>
<tr>
<td>VLAT [80]</td>
<td>512</td>
<td>Hamming</td>
<td>55.0</td>
<td>-</td>
<td>33.0</td>
<td>-</td>
</tr>
<tr>
<td>FV [35]</td>
<td>512</td>
<td>Hamming</td>
<td>50.0</td>
<td>-</td>
<td>30.0</td>
<td>2.79</td>
</tr>
<tr>
<td>FV [35]</td>
<td>512</td>
<td>Asymmetric</td>
<td>56.9</td>
<td>-</td>
<td>35.1</td>
<td>3.10</td>
</tr>
<tr>
<td>VLAD\textit{our}</td>
<td>4k</td>
<td>Hamming</td>
<td>55.6</td>
<td>54.5</td>
<td>-</td>
<td>3.16</td>
</tr>
<tr>
<td>VLAD\textit{our}</td>
<td>1k</td>
<td>Hamming</td>
<td>51.0</td>
<td>49.1</td>
<td>33.5</td>
<td>3.00</td>
</tr>
<tr>
<td>VLAD\textit{our}</td>
<td>512</td>
<td>Hamming</td>
<td>49.5</td>
<td>45.9</td>
<td>30.8</td>
<td>2.92</td>
</tr>
<tr>
<td>FV\textit{our}</td>
<td>4k</td>
<td>Hamming</td>
<td>58.4</td>
<td>55.4</td>
<td>-</td>
<td>3.2</td>
</tr>
<tr>
<td>FV\textit{our}</td>
<td>1k</td>
<td>Hamming</td>
<td>54.7</td>
<td>50.2</td>
<td>36.5</td>
<td>3.07</td>
</tr>
<tr>
<td>FV\textit{our}</td>
<td>512</td>
<td>Hamming</td>
<td>52.2</td>
<td>46.9</td>
<td>32.7</td>
<td>2.95</td>
</tr>
<tr>
<td>RVD-W</td>
<td>4k</td>
<td>Hamming</td>
<td>\textbf{61.3}</td>
<td>\textbf{57.7}</td>
<td>-</td>
<td>\textbf{3.25}</td>
</tr>
<tr>
<td>RVD-W</td>
<td>1k</td>
<td>Hamming</td>
<td>\textbf{59.1}</td>
<td>\textbf{53.9}</td>
<td>\textbf{40.8}</td>
<td>\textbf{3.18}</td>
</tr>
<tr>
<td>RVD-W</td>
<td>512</td>
<td>Hamming</td>
<td>\textbf{57.9}</td>
<td>\textbf{50.6}</td>
<td>\textbf{36.6}</td>
<td>\textbf{3.10}</td>
</tr>
<tr>
<td>FV\textit{our}</td>
<td>4k</td>
<td>WH</td>
<td>63.0</td>
<td>58.0</td>
<td>-</td>
<td>3.30</td>
</tr>
<tr>
<td>FV\textit{our}</td>
<td>1k</td>
<td>WH</td>
<td>60.0</td>
<td>53.1</td>
<td>38.0</td>
<td>3.18</td>
</tr>
<tr>
<td>FV\textit{our}</td>
<td>512</td>
<td>WH</td>
<td>55.4</td>
<td>47.5</td>
<td>33.5</td>
<td>3.06</td>
</tr>
<tr>
<td>RVD-W</td>
<td>4k</td>
<td>WH</td>
<td>\textbf{67.6}</td>
<td>\textbf{60.3}</td>
<td>-</td>
<td>\textbf{3.40}</td>
</tr>
<tr>
<td>RVD-W</td>
<td>1k</td>
<td>WH</td>
<td>\textbf{63.5}</td>
<td>\textbf{55.2}</td>
<td>\textbf{43.6}</td>
<td>\textbf{3.31}</td>
</tr>
<tr>
<td>RVD-W</td>
<td>512</td>
<td>WH</td>
<td>\textbf{61.6}</td>
<td>\textbf{51.1}</td>
<td>\textbf{38.8}</td>
<td>\textbf{3.22}</td>
</tr>
</tbody>
</table>

4.4 Binary RVD versus the state of the art
This section culminates our research by combining the RVD framework with other novel elements proposed in this chapter, and evaluating the results. Table 4.1 demonstrates the performance of the RVD-W to the latest state-of-the-art, when compact binary representations are used. It can be clearly seen that RVD-W outperforms all global signatures on all datasets. Using Hamming distance as a similarity measure, the RVD-W signature (4k bits) improves significantly (gain of +15% mAP) over BoW (20k bits) and also over FV \cite{88} (gain of +3.4%) on the Holidays dataset. Compared to the recent VLAT method (512 bits), the RVD-W (512 bits) offers a gain of +3% on Holidays dataset and +3.6% on large scale dataset of Holidays1M. The RVD-W also outperforms FV \cite{35} which employs Asymmetric distance (AD) measure to compute similarity between two FVs. The RVD-W provides superior performance to VLAD\textsubscript{our} and FV\textsubscript{our} on all datasets and at each bitrate. FV\textsubscript{our} denotes our implementation of FV, which performs better than the original FV \cite{88,35}, primarily due to conversion of SIFT vectors to RootSIFT and application of power normalization on the FV.

Using Weighted Hamming (WH) distance as a similarity measure, the RVD-W brings a further dramatic improvement over all prior-art methods. Compared to FV (our implementation), the RVD-W (4k bits) provides an improvement of +4.5% and +2.3% on Holidays and Oxford5k datasets. The difference in retrieval performance is even more significant on large scale dataset of Holidays1M (+5.6%) compared to FV our (1k bits). On UKB, RVD-W (4k bits) obtains a recall@4 of 3.40 which significantly exceeds any results published to date.

4.5 Conclusion

In this chapter, we show how to encode high-dimensional and floating point global descriptors to compact binary codes using several techniques including PCAE, PCAE+RR, and PCAE+ITQ with iterative Quantization. The detailed evaluation on the retrieval datasets confirms that the core RVD-W signature outperforms FV and VLAD also in the binary domain. We also show that applying rotation after PCA (PCAE+RR and PCAE+ITQ) significantly deteriorates the retrieval performance due to the fact that the de-correlation performed by the PCA is lost. We present a novel binary match-
ing algorithm which combines PCAE with weighted matching where the high variance directions are given more weight compared to low-variance directions. We derive theoretically the optimum values for the weights and show a dramatic improvement in performance.
Chapter 5

Mobile Visual Search based on Scalable RVD

Mobile phone has become a very powerful instrument with faster CPUs and GPUs, larger memory capacity, high resolution cameras, and more accurate sensors including GPS transceivers and gyroscopes. These mobile hardware technologies help to accelerate exciting applications of computer vision in mobile scenarios. One of the most prominent example of such applications is mobile visual search (MVS). MVS applications enable user to interact with objects 'seen' by the camera and retrieve corresponding information and metadata about the objects. A consumer can, for example, use their mobile phones to recognize and compare prices of products such as books, DVDs and clothes, or a tourist can take a photo of a landmark and receives meaningful information such as its name, location or history from an on-line database.

Mobile visual search systems require algorithms with high recognition performance, supporting scalable and compact bitstream representations. Additional requirements include: (1) low computational complexity to improve execution speeds and battery life, and (2) low system memory footprint to reduce silicon costs. ISO/MPEG is currently standardising Compact Descriptors for Visual Search (CDVS) and it has formulated following requirements based on inputs from industry:

- **Robustness:** The MVS application should achieve high matching accuracy for
images of various categories: graphic objects, buildings, video frames, common objects and paintings.

- **Hardware implementation efficiency:** The CDVS standard requires algorithms with low computational complexity to improve execution speeds and battery life and small memory footprint for low-cost hardware implementations.

- **Support web-scale databases:** A global signature should be extracted from each image to facilitate large scale search. The global descriptor can be matched quickly with database global descriptors, thus quickly generating a short list of images for the geometric verification step.

- **Scalability:** The image signature extracted on query side should be scalable with the size ranging from 512B to 16kB per image. The scalability is important as it provides trade-off between performance and size, required for variable channel bandwidth.

In this chapter, we address all these technical challenges simultaneously by developing a compact and scalable global binary descriptor based on the core RVD. Furthermore, we design and develop an efficient MVS system which delivers beyond state-of-the-art performance. Our main contributions include:

- We introduce bit-rate scalability in the RVD framework by employing Cluster Selection (CS) and Bit Selection (BS) mechanisms to support interoperable binary image representations that can be easily adapted to the requirements of the use scenario, for example communication channel bandwidth or storage limitation.

- We propose a very simple, fast and effective score function based on weighted Hamming distance, to compute similarity between two binary representations.

- We optimise the entire MVS framework and compare our system performance to the MPEG CDVS reference model.

This chapter is organized as follows: In Section 5.1 we describe the structure of a modern mobile visual search system. Section 5.2 presents the architecture and evaluation
5.1 Modern Mobile Visual Search System

Figure 5.1: Mobile Visual Search System

framework of the MPEG CDVS standard. In Section 5.3 we present the design of our scalable Robust Visual descriptor (RVD). The experimental setup and the detailed evaluation of our method is presented in Section 5.4. Finally, in Section 5.5 we compare our results with the state-of-the-art demonstrating significant improvement over REVV and SCFV on the MPEG CDVS datasets.

5.1 Modern Mobile Visual Search System

Figure 5.1 shows an example of a typical Mobile Visual Search system. The camera on a mobile device captures a photo or video of a scene and the system extracts local image features. The local descriptors are then aggregated to form global image signature to enable fast matching. The local features and global descriptors are encoded and sent to the remote server over a wireless link. On the server side, the decoding algorithm is performed and the query global descriptor is then matched against the pre-extracted descriptors from images stored in the remote database. Based on the similarity score, the database images are ranked and the top matches are passed to the geometric verification step. Finally, the most similar images are sent back to the mobile device.
5.2 Compact Descriptors for Visual Search (CDVS) standard

CDVS [4] is the standard recently approved by ISO/ MPEG, in order to provide a practical, inter-operable, effective and cross-platform solution for visual search applications. This standard is geared towards applications such as identifying 3D objects, outdoor landmark recognition, mobile commerce, finding information about CDs, books, artworks and printed documents. The main aim of the CDVS standard is to develop an optimized image representation and to simplify the design of image matching applications.

This section briefly presents architecture of the CDVS standard. First we present the extraction and encoding of local and global descriptors. Then we describe the CDVS evaluation framework aimed to evaluate the performance of a visual search system.

5.2.1 Compact descriptor extraction

The compact descriptor of an image comprises two components, namely a set of compressed local descriptors with their locations and a global descriptor, representing the content of the entire image. Figure 5.2 shows the CDVS pipeline used to extract a compact descriptor from an image [4]. Extraction consists of the following steps:

1. Keypoints detection. The keypoints are detected in an image based on the Laplacian-of-Gaussian scale-space and determination of extrema by means of polynomial approximations.
2. **Feature selection.** The feature selection method selects the keypoints with high matching probabilities based on several factors such as keypoints scale, coordinates, and orientation.

3. **Local Descriptor Extraction.** Extraction of SIFT descriptor based on the spatial distribution of pixel intensity gradients in a scale and orientation normalized patch surrounding the keypoint.

4. **Local Descriptor Compression:** Compression of the SIFT descriptors using a specially designed transform which computes the sums and differences of different SIFT gradient bins, and selects elements following a specific pattern to preserve the discriminative power of SIFT.

5. **Coordinate Coding.** The keypoint coordinates are encoded using Location Histogram Coding (LHC) technique [111], where the coordinate information is converted into a histogram and a context adaptive arithmetic encoder is used to compress the histogram.

6. **Global descriptor aggregation.** Aggregating of local image descriptors into a compact and binary global signature such as VLAD, Fisher Vector or RVD.

### 5.2.2 CDVS evaluation framework [4]

The CDVS evaluation framework is designed to test two distinct tasks performed by visual search systems: retrieval and pairwise matching. The former addresses retrieval of images depicting instances of a user specified query object from a large database of images. The latter verify whether the query and the reference image contains the same object.

### 5.2.3 Retrieval architecture

Figure 5.3 depicts the retrieval process. Each image in the database is represented by a global descriptor and a set of $T$ local descriptors (typically $T = 300$) with their locations. Now given a query, a global descriptor is first computed by aggregating local
5.2. Compact Descriptors for Visual Search (CDVS) standard

Descriptors and compared against the database of pre-computed global descriptors. Based on the similarity scores, the database images are ranked and the top $L$ candidates (typically 500) are forwarded to the local feature matching and the geometric re-ranking steps. The short-listed database images are re-ranked based on the number of inliers computed by RANSAC algorithm, which reflects the number of matching descriptor pairs which are geometrically consistent.
5.3. Scalable Robust Visual descriptor (RVD)

5.2.4 Pairwise matching architecture

Figure 5.4 describes the Pairwise Matching (PM) process, which compares the descriptors extracted from two images and returns matching/non-matching decision. The PM process encompasses a comparison between the global representations of the query and the reference images, followed by the matching of the local descriptors extracted from both images.

For the matching of global descriptors, given two binary signatures $R_X$ and $R_Y$ extracted from images $X$ and $Y$, the similarity score is based on a weighted correlation between the binary descriptors. If the similarity score is greater than a certain threshold then this image pair is considered to be matching. More detail about global descriptor matching process is presented in section 5.3.4.

The local descriptors and their corresponding coordinates are decoded for both the query and the reference image. A set of putative matches is computed, each one comprising of a pair of local descriptors, one in each image, that pass the Lowe [70] distance ratio test. The RANSAC algorithm is applied on the putative descriptor matches to obtain a set of inlier point pairs. If the number of inliers are greater than a certain threshold then the query and the reference image are considered to be matching.

5.3 Scalable Robust Visual descriptor (RVD)

In this section we introduce a Mobile Visual Search system based on Robust Visual Descriptor (RVD). We have modified RVD so that it delivers a scalable representation with a small memory footprint yet yielding highly accurate retrieval and pairwise matching performance. It also scales well, supporting fast searches through large image databases. Figure 5.5 depicts the RVD pipeline for extraction and matching of global descriptors.

Compared to the RVD binary representation outlined in Section 4.2, we have introduced cluster selection and bit selection modules and improved matching referred to as superior matching.
5.3. Scalable Robust Visual descriptor (RVD)

Local descriptor Extraction

Given an image, SIFT features are extracted and selected to generate the RVD descriptor. The SIFT descriptors are first L1-normalized and then power normalised with the factor 0.5. The dimensionality of SIFT features is reduced from 128-dim to $d'$ dimensions using PCA. The PCA parameters are calculated off-line based on approximately 5 million SIFT features, extracted from a dataset independent of the MPEG evaluation dataset.

Robust Rank-based aggregation

The aggregation follows our core design as presented in Section 3.1 which is summarised here for completeness.

Each local feature $x_t$ is assigned to $K$-nearest clusters ($\text{NN}_\gamma^K$), where $\{\gamma = 1...K\}$ denotes the rank of a particular nearest cluster and $K$ is the maximum rank considered. For each cluster $\mu_j$ with rank $\gamma$ and each associated local descriptor, the residual vectors $x_t - \mu_j$ are computed and subsequently L1-normalized. The normalized residual vectors are then weighted based on rank assignment weights $\tau_{tj}$ before aggregation into cluster...
5.3. **Scalable Robust Visual descriptor (RVD)**

![Image 1](image.png)

**Figure 5.6:** The reliability scores $CS_j$ for a image 1 using $n = 170$ cluster centres

level representation $\zeta_j$. The RVD $R$ signature is formed by stacking all $\zeta_j$ for $j = 1, \ldots, n$ (refer Section 3.1).

### RVD with local Whitening (RVD-W)

In this variant of RVD (Section 3.2), the weighted residual vector $r_{tj}$ are transformed via a local PCA matrix $P_j$ and subsequently whitened before aggregation into cluster level representation $\zeta_j$. The $\zeta_j$ vectors are concatenated to form the final RVD-W representation $R^w$.

#### 5.3.1 Component Cluster selection

This section presents a new element designed to enable scalability of the resulting RVD descriptor. Scalable RVD is formed by concatenating a selected subset of cluster-level component descriptors $\zeta_j$, chosen based on their expected reliability. More precisely, the reliability factor $CS_j$ of each cluster is computed, based on the number of local descriptors associated with that cluster at each rank $\gamma$ as follows:

$$CS_j = \sum_{\gamma=1}^{K} \sum_{x_t:NNF(x_t) = j} \tau_{tj} \quad (5.1)$$
where $\tau_{tj}$ are weights associated with particular ranks $\gamma$. In RVD, the following weights are employed: $\tau_{tj} = 1$ if $NN_1^K(x_t) = j$, $\tau_{tj} = 0.5$ if $NN_2^K(x_t) = j$, $\tau_{tj} = 0.25$ if $NN_3^K(x_t) = j$ and $\tau_{tj} = 0$ otherwise.

Our selection is based on the observation that clusters with low level of occupancy are often affected by presence of outliers. Therefore, we remove clusters with low occupancy to obtain a discriminative and scalable RVD representation. A particular cluster is rejected if its reliability score $CS_j$ is less than cluster selection threshold $CS_{th}$. The threshold values are selected to achieve the required size of the RVD representation for each bit-rate requested by the CDVS framework. Figure 5.6 illustrates the reliability scores $CS_j$ for a particular image where a RVD representation with $n = 170$ cluster centres is used. It can be observed that several clusters have a significantly low $CS_j$ values and therefore will be rejected. An example of the cluster selection mechanism employed in RVD aggregation is shown in Figure 5.7.

### 5.3.2 Binarizing RVD

Having obtained a shorter and more robust image signature using cluster selection mechanism, we can now binarize the signature based on the sign of the $\zeta_j$ coefficients:

$$\zeta_j^b = [\text{sign}\{\zeta_j^1\}; \text{sign}\{\zeta_j^2\}; \ldots; \text{sign}\{\zeta_j^d\}]$$  (5.2)

The $\zeta_j^b$ vectors are concatenated to form the binary RVD vector $R^b$. Similarly, the RVD-W global descriptor $R^w$ is compressed using sign binarization to form vector $R^{wb}$.

### 5.3.3 Bit Selection in RVD

To achieve an even more compact representation a subset of elements of the aforementioned binary representations can be selected. There are many strategies to select 'more informative bits' and we have suggested and investigated four possible strategies:

1. **Top selection (TSel)**: We select top $l$ bits from each cluster level representation $\zeta_j^b$. The selected bits are stacked to from binary global descriptor $R^b$. 

5.3. Scalable Robust Visual descriptor (RVD)

\[ C_{th} = 10 \]
\[ CS_1 = 4 \times 3 + 2 \times 2 + 1 \times 1 = 17 \text{ Accept} \]
\[ CS_2 = 4 \times 1 + 2 \times 1 + 1 \times 2 = 8 \text{ Reject} \]
\[ CS_3 = 4 \times 1 + 2 \times 2 + 1 \times 1 = 9 \text{ Reject} \]
\[ CS_4 = 4 \times 4 + 2 \times 1 + 1 \times 0 = 18 \text{ Accept} \]

Figure 5.7: An illustration of the cluster selection approach employed in RVD aggregation. The polygon indicates a Voronoi cell. The descriptors \( x_1, x_2 \) and \( x_3 \) (shown as grey dots) belong to \( NN^K_1 \) of cluster centre \( \mu_1 \). The descriptors \( x_4, x_5 \) (shown as green dots) and \( x_6 \) (shown as yellow dot) belong to \( NN^K_2 \) and \( NN^K_3 \) of \( \mu_1 \) respectively. The reliability factor \( CS_1 \) of \( \mu_1 \) calculated according to equation 5.1 is 17. The occupancy threshold \( CS_{th} = 10 \) is used to achieve the required size of the RVD representation for bitrate 1k (refer to Section 5.4.4). Since the factor \( CS_1 \) is greater than the \( CS_{th} = 10 \), cluster 1 is used in RVD aggregation. Conversely, cluster 2 has reliability factor 8 and is therefore rejected.
5.3. **Scalable Robust Visual descriptor (RVD)**

Figure 5.8: Distribution of Hamming distances between $\zeta^b_j$ vectors at codewords 1 and 64 for matching and non-matching image pairs, using four Bit selection methods: Top selection (TSel), Random selection (NSel), selection based on difference (Dsel) and selection based on ratio (RSel)
2. **Random selection (Nsel):** We randomly select \( l \) bits from each cluster level representation \( \zeta^b_j \) to form signature \( R^b \).

3. **Selection based on difference (Dsel):** We select those bits from each \( \zeta^b_j \) which provide best separability between hamming distances for matching and non-matching pairs (trained off-line). More precisely, let \( Pr(\chi|m) \) and \( P(\chi|nm) \) denote the conditional probability that the XOR between two corresponding bits is 1 for matching and non-matching image pairs respectively. We select bits that maximise the difference between \( Pr(\chi|m) \) and \( Pr(\chi|nm) \).

4. **Selection based on ratio (Rsel):** We select those bits from each \( \zeta^b_j \) that maximise the log ratio between \( Pr(\chi|m) \) and \( Pr(\chi|nm) \).

**Analysis of Bit Selection methods**

We will use the separability between matching and non-matching images to predict the performance of the aforementioned bit selection methods. More precisely, let us denote \( Pr(h|m) \) and \( Pr(h|nm) \) as the probability of observing a Hamming distance \( h \) at any codeword for a matching and non-matching image pair respectively. Figure 5.8 shows the distribution of \( Pr(h|m) \) and \( Pr(h|nm) \), for codewords 1 and 64, using aforementioned Bit selection methods. The distance between the non-matching and matching distributions is computed using KL-divergence [60]:

\[
\text{dist} = \sum_{i=1}^{l} Pr(h|m)(i) \ln \frac{Pr(h|m)(i)}{Pr(h|nm)(i)}
\]  

where \( l \) is the number of bits in \( \zeta^b_j \). A few important observations can be made:

- The DSel method provides the best separability (maximum KL-Divergence) between matching and non-matching distributions for both codewords.

- For Nsel method, the area of overlap between matching and non-matching distributions is maximum, predicting lower performance.

Table 5.1 presents the mean and median values of KL-divergence across \( n = 170 \) codewords, for four different Bit-selection methods. It can be observed that the best separation between non-matching and matching distributions is achieved using DSel method.
5.3. **Scalable Robust Visual descriptor (RVD)**

Based on the KL-divergence, we expect the bits selected by the Dsel method to perform the best in pairwise matching and retrieval experiments. This has been confirmed by experiments as shown in Figure 5.10 (Section 5.4.5).

Table 5.1: Mean and median of KL-divergence scores across 170 codewords using four Bit selection methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSel</td>
<td>1.55</td>
<td>1.53</td>
</tr>
<tr>
<td>NSel</td>
<td>1.37</td>
<td>1.32</td>
</tr>
<tr>
<td>DSel</td>
<td>1.99</td>
<td>1.97</td>
</tr>
<tr>
<td>RSel</td>
<td>1.68</td>
<td>1.63</td>
</tr>
</tbody>
</table>

### 5.3.4 RVD matching

We employ a very fast matching algorithm based on the Hamming distance. Given two images, the similarity score is a weighted correlation between their binary signatures and can be calculated effectively with bitwise XOR and POPCNT instructions. The Hamming distance is then used as index to read weights, which are stored in a small look-up table. If a Hamming distance cannot be computed, due to missing cluster-level sub-descriptors for one or both images, a fixed penalty $\vartheta$ is assigned. This is indicated by $\Phi_{1,j} \times \Phi_{2,j}$ in Equation 5.4, where $\Phi_{1,j} \times \Phi_{2,j} = 1$ only when both images contain the $j^{th}$ cluster binary representation.

Let $\zeta_{1,j}^b$ and $\zeta_{2,j}^b$ be the binary vectors for two images at $j^{th}$ codeword. The overall correlation between two images is calculated as:

$$Sc = \sum_{j=1}^{n} \Phi_{1,j} \Phi_{2,j} W(h(\zeta_{1,j}^b, \zeta_{2,j}^b)) + \vartheta(1 - \Phi_{1,j} \Phi_{2,j})$$  \hspace{1cm} (5.4)

where $h(.,.)$ denotes the Hamming distance and $W$ denotes the weights to the Hamming distance. Weights $W$ are learned from the matching/non-matching image pairs drawn from an independent dataset. The weighting function $W$ for each cluster $j$ is computed as follows:

$$W = \log \frac{Pr(h|m)}{Pr(h|nm)}$$  \hspace{1cm} (5.5)
In the RVD matching scheme, we used a single weighting function for all clusters to reduce the matching time and memory requirements. This function is computed by taking a median of weights for each cluster.

In Equation 5.4, the value for penalty $\vartheta$ is set to -0.2, reflecting the expectation that misaligned representations indicate mismatch between images. This value was found to perform well for all bitrates and for both pairwise matching and retrieval scenarios alike.

5.4 Experimental Setup and Evaluation

The purpose of this section is to evaluate the RVD mobile visual search system in context of the other state-of-the-art descriptors developed for MPEG. We first introduce the datasets and protocols used to evaluate the performance and then analyse the impact of the novel components that constitute our method, namely application of local PCA with whitening, cluster selection and bit selection methods. A comparison of different MVS systems described in section 5.3 is presented at the end of this section.

5.4.1 Datasets

The performance of our method is extensively evaluated using MPEG CDVS test protocols. The CDVS evaluation consists of pairwise image matching and retrieval experiments. Five image categories are used (1) Text and graphics including Book/DVD covers/documents/business cards, (2) Photographs of Paintings, (3) Video frames, (4) Landmarks and (5) Common objects from the UKB dataset. Example of images in MPEG dataset are shown in Figure 5.9.

The CDVS test protocol is designed to test two distinct tasks performed by visual search systems: pairwise matching (PM) and retrieval.

The pairwise matching determines whether the query and the reference image is matching (contain same object) or non-matching. MPEG has a dataset of matching image pairs (16319 pairs) and a dataset of non-matching image pairs (171,815 pairs) from
all CDVS categories (Table 5.2). The PM performance is measured in terms of True Positive Rate (TPR) at a given False Alarm Rate (FPR). The MVS system strive to maximize TPR at a given FPR level, defined as less than 1%.

The retrieval experiments aim to accurately match query images against a large set of database images. A dataset of 18,440 images gathered from the five aforementioned categories is augmented with 1 million distractor images, to form the retrieval database. A total of 11,313 queries are used to evaluate the MVS system performance. Table 5.3 presents the number of queries and the database images contained in each of the five categories. Retrieval accuracy is measured by mean Average Precision (mAP).

### 5.4.2 Local descriptor extraction

In a particular image, key-points are detected using the Block-based Frequency Domain Laplace of Gaussian [4] and local regions are encoded in a 128-dimensional SIFT descriptor [75]. Typically 300 SIFT features per image are retained at each operating point using the default CDVS feature selection mechanism [4], when less than 300 features are present all of them are selected. The selected SIFT descriptors are converted
Table 5.2: Number of matching and non-matching image pairs for pairwise matching experiments

<table>
<thead>
<tr>
<th>CDVS category</th>
<th>Total number of images</th>
<th>Number of matching pairs</th>
<th>Number of non-matching pairs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graphics</td>
<td>7500</td>
<td>9000</td>
<td>90000</td>
</tr>
<tr>
<td>Paintings</td>
<td>455</td>
<td>364</td>
<td>3640</td>
</tr>
<tr>
<td>Video frames</td>
<td>500</td>
<td>400</td>
<td>4000</td>
</tr>
<tr>
<td>Buildings</td>
<td>14935</td>
<td>4005</td>
<td>48675</td>
</tr>
<tr>
<td>Objects</td>
<td>10200</td>
<td>2550</td>
<td>25500</td>
</tr>
<tr>
<td>Total</td>
<td>33590</td>
<td>16319</td>
<td>171815</td>
</tr>
</tbody>
</table>

Table 5.3: Number of query images and database images for retrieval experiments

<table>
<thead>
<tr>
<th>CDVS category</th>
<th>Total number of images</th>
<th>Query images</th>
<th>Database images</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graphics</td>
<td>7500</td>
<td>4500</td>
<td>1000</td>
</tr>
<tr>
<td>Paintings</td>
<td>455</td>
<td>364</td>
<td>91</td>
</tr>
<tr>
<td>Video frames</td>
<td>500</td>
<td>400</td>
<td>100</td>
</tr>
<tr>
<td>Buildings</td>
<td>14935</td>
<td>4005</td>
<td>9559</td>
</tr>
<tr>
<td>Objects</td>
<td>10200</td>
<td>2550</td>
<td>7690</td>
</tr>
<tr>
<td>Total</td>
<td>33590</td>
<td>11313</td>
<td>18440</td>
</tr>
</tbody>
</table>

to RootSIFT [7] without any additional storage or memory. The SIFT descriptors are projected to 48 dimensional space using PCA. The size of codebook is fixed at 170.

5.4.3 Memory footprint of RVD, SCFV and REVV

The CDVS group has been placing a significant effort to reduce the memory requirements of the CDVS pipeline. This is to support low-cost hardware implementation of the CDVS compliant systems. Table 5.4 compares the memory requirement of RVD, SCFV and REVV.
Table 5.4: Memory footprint of RVD, SCFV and REVV (KM:k-means; GMM: Gaussian Mixture Model; BSel: Bit selection)

<table>
<thead>
<tr>
<th>Method</th>
<th>SIFT (PCA)</th>
<th>Vocabulary</th>
<th>Descriptor transform</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>REVV</td>
<td>-</td>
<td>24kB (KM)</td>
<td>95kB (LDA)</td>
<td>119kB</td>
</tr>
<tr>
<td>SCFV</td>
<td>17kB</td>
<td>33kB (GMM)</td>
<td>-</td>
<td>49kB</td>
</tr>
<tr>
<td>RVD</td>
<td>6kB</td>
<td>8kB (KM)</td>
<td>2kB (BSel)</td>
<td>16kB</td>
</tr>
</tbody>
</table>

1. **RVD** auxiliary data consists of (i) the SIFT PCA matrix $128 \times 48$ (1 byte per element), plus a 128-dimensional mean vector (1 byte per dimension), (ii) the table containing Cluster Centres is $170 \times 48$ elements (1 byte per element), and (iii) a bit selection table $2 \times 170 \times 48$ bits.

2. **SCFV** memory requirements include (i) the SIFT PCA matrix $128 \times 32$ plus a 128-dimensional mean vector (4 byte per element), and (ii) the GMM parameters involves a set of $\{\omega_j, \mu_j, \Sigma_j\}$ for Gaussian $j$ resulting in $128 \times (1 + 32 + 32)$ (4 byte per element).

3. **REVV** auxiliary data consists of (i) Codebook $190 \times 32$ (4 bytes per element), and (ii) cluster level LDA coefficients $190 \times 32 \times 20$ elements (1 byte per element).

It can be seen that RVD has the smallest footprint.

### 5.4.4 Scalable RVD

To evaluate the descriptor size scalability, CDVS standard requires Mobile Visual Search systems to present results at 6 target operating points with different query size budgets: 512 bytes, 1 KB, 2 Kilobytes, 4 kb, 8 kb and 16 kb. The same limit is also placed on the descriptors stored in the database.

The maximum size of RVD descriptor is 1 kilobyte which can be scaled down to any required bitrate via cluster selection and bit selection mechanisms. In MPEG CDVS
three interoperable RVD sizes are used: 279B, 328B and 756B, which after addition of the compressed local SIFT descriptors and their locations creates descriptor with overall size between 512B and 16kB. Table 5.5 presents different parameters used to produce scalable RVD global signature. It can be observed that there is no bit selection for the highest descriptor length of 4k, 8k, and 16k, where all 48 bits are used. For 1k and 2k descriptor length, 32 bits are selected and 24 bits are selected for 512 bit descriptor length. In RVD, a cluster occupancy threshold $CS_{th}$ is applied:

<table>
<thead>
<tr>
<th>Global descriptor size</th>
<th>Bits per cluster</th>
<th>Number of clusters selected</th>
<th>Cluster selection threshold $CS_{th}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>512 B</td>
<td>279 B</td>
<td>24</td>
<td>8</td>
</tr>
<tr>
<td>1 kB</td>
<td>328 B</td>
<td>32</td>
<td>10</td>
</tr>
<tr>
<td>2 kB</td>
<td>328 B</td>
<td>32</td>
<td>10</td>
</tr>
<tr>
<td>4 kB</td>
<td>756 B</td>
<td>48</td>
<td>4</td>
</tr>
<tr>
<td>8 kB</td>
<td>756 B</td>
<td>48</td>
<td>4</td>
</tr>
<tr>
<td>16 kB</td>
<td>756 B</td>
<td>48</td>
<td>4</td>
</tr>
</tbody>
</table>

is rejected if $CS_j < CS_{th}$. The threshold values are selected to achieve the required size of the RVD representation for each bitrate. At bitrates 16kB, 8kB and 4kB, about 126 clusters are selected by applying occupancy threshold $CS_{th} = 4$, while at 2kB and 1kB 82 clusters are selected ($CS_{th} = 10$). At lowest bitrate of 512B, we set $CS_{th} = 8$. Similar parameters are also employed in RVD-W formation.

### 5.4.5 Bit Selection methods

In this section we compare the performance of different bit selection methods discussed in Section 5.3.3 specifically Top selection (TSel), Random selection (NSel), selection based on difference (DSel) and selection based on ratio (RSel). Our previous evaluation exploring the separability between the matching and non-matching distributions using KL-divergence, indicated that DSel method is likely to provide best results. It can be clearly seen from Figure 5.10 that the DSel method indeed delivers the best performance
Figure 5.10: Pairwise matching performance using four Bit Selection methods: Top selection (Tsel), Random selection (Nsel), selection based on difference (Dsel) and selection based on ratio (Rsel)
on all datasets. Therefore, in all the following experiments, we used the DSel method to select bits from each cluster level representation \( \zeta_j \) to form scalable RVD descriptor.

5.5 Comparison with the state of the art

We compare the performance of the proposed method to the latest state-of-the-art, Residual Enhanced Visual Vector (REVV) and Scalable Compressed Fisher Vector (SCFV) in two scenarios, when only global descriptor is used and when global descriptor is combined with local descriptors.

5.5.1 Comparison in global descriptor matching

In this section we evaluate the performance of the global representations RVD, RVD-W and SCFV. All the results are computed by performing global descriptor matching only without exploiting local descriptor matching nor geometric verification. We first present the pairwise matching results calculated as the True Positive Rate (TPR) at a given False Alarm Rate (FPR). To ensure a fair comparison between different methods, the FAR at each bitrate is kept the same. It can be seen from the Figure 5.11 that both RVD and RVD-W significantly outperform SCFV on all datasets and at each bitrate. Compared to SCFV, RVD offers an average gain of +9% on planar 2D objects datasets (Graphics, Paintings and Video Frames). The average difference in pairwise matching performance is even more significant (+12%) on more challenging 3D objects datasets (Buildings and Objects). The best representation, RVD-W, provides an average improvement of +11% and +15% on 2D and 3D objects datasets respectively, over SCFV.

Figure 5.12 presents the retrieval performance of RVD, RVD-W and SCFV on a large scale dataset (1M). The accuracy is measured by mean Average Precision (mAP). It can be observed that RVD delivers an average gain of 1% and 3.6% on 2D and 3D objects datasets over SCFV. The RVD-W representation achieves an even more dramatic improvement of 3.5% and 4.5% compared to SCFV.
Figure 5.11: Pairwise matching using global descriptors: RVD, RVD-W and SCFV on (a) Graphics, (b) Paintings, (c) Video frames, (d) Buildings, and (e) Objects (all results in TPR(%)
5.5. Comparison with the state of the art

Figure 5.12: Image retrieval using global descriptors: RVD, RVD-W and SCFV on (a) Graphics, (b) Paintings, (c) Video frames, (d) Buildings, and (e) Objects (all results in mAP(%));
5.5. Comparison with the state of the art

5.5.2 Comparison in global and local descriptor matching

RVD and RVD-W global descriptors are integrated into the MPEG CDVS standard to perform full evaluation. In all the experiments both local and global image descriptors are used to perform matching between images, as explained in Section 5.2. We first present the pairwise matching results using four mobile visual search systems: RVD, RVD-W, SCFV and REVV. Firstly, it can be observed from Figure 5.13 that the performance of all systems is very good and already saturated for 2D objects (Graphics, Paintings and Video Frames). However for 3D non-planar objects (Buildings and Objects), both RVD and RVD-W significantly outperform SCFV and REVV systems.

Figure 5.14 illustrates the retrieval performance of RVD, RVD-W, SCFV and REVV where a short list of images retrieved by global descriptor is re-ranked using local descriptors including geometric verification. It can be clearly seen that RVD and RVD-W significantly outperform REVV on all datasets and at every bitrate. Compared to SCFV, RVD-W provides a gain of 1.5% and 3.5% on 2D objects and 3D objects datasets.

5.5.3 Overall comparison of Mobile Visual Search systems performance

In this section, we compute an average pairwise matching and retrieval performance over all experiments and at all bitrates using global descriptors only without exploiting geometric verification. It can be seen from Figure 5.15(a) and Figure 5.15(b) that both RVD and RVD-W significantly outperform the SCFV global descriptor. The RVD-W system provides an average gain of 12% in pairwise matching and 4% in retrieval, over SCFV.

Now we perform the full evaluation of various MVS systems performance when integrated into the MPEG CDVS system. Figure 5.15(c) and Figure 5.15(d) illustrate the improvement offered by the RVD and RVD-W systems compared to REVV and SCFV. The RVD-W system provides an average gain of 1.7% in TPR and 7.5% in mAP over REVV. Compared to the state-of-the-art SCFV system RVD-W provides a gain of 1.7% in both pairwise matching and retrieval experiments.
5.5. Comparison with the state of the art

Figure 5.13: Pairwise matching performance of RVD, RVD-W, SCFV and REVV on (a) Graphics, (b) Paintings, (c) Video frames, (d) Buildings, and (e) Objects (all results in TPR(%));
5.5. Comparison with the state of the art

Figure 5.14: Retrieval performance of RVD, RVD-W, SCFV and REVV on (a) Graphics, (b) Paintings, (c) Video frames, (d) Buildings, and (e) Objects (all results in mAP(%));
5.6 Conclusion

A novel global image descriptor is proposed which combines rank-based assignment with robust aggregation framework and performs a mid-stage de-correlation & whitening of the residual vectors. The scalability of global signature is achieved by applying two novel methods: Bit Selection and Cluster Selection. Extensive experiments demonstrate excellent recognition performance, outperforming the latest state-of-the-art algorithms with binary representations. RVD and RVD-W significantly outperform the REVV and SCFV methods in both pairwise matching and retrieval, with exceptional improvements.

Figure 5.15: Performance of RVD, RVD-W, SCFV and REVV within the CDVS framework TM7 (a) Pairwise matching results using global descriptors, (b) Retrieval results using global descriptors, (c) Pairwise matching results using global and local descriptors and (d) Retrieval results using local and global descriptors.
in the cases of 3D non-planar objects. The extraction process requires low memory and matching is very fast, conforming to MPEG CDVS requirements.
Chapter 6

Image Classification based on higher order RVD

As humans, we are able to recognize categories of objects and scenes effectively and effortlessly. Moreover learning new categories require very little supervision and typically only a few examples. The aim of the computer vision research is to develop algorithms that can ultimately achieve the same level of recognition performance. Such recognition or classification capabilities can support a great number of useful applications including web search, organisation of photo/video libraries, surveillance, biometrics, robotic vision etc. The task of performing accurate and scalable classification is challenging mostly due to large intra-class visual diversity, significant similarities between classes, background clutter and partial occlusions.

A classification system, illustrated in Figure 6.1 typically consists of three blocks [21]: (1) local feature extraction, (2) derivation of image representation via aggregation, and (3) image classification. Generally, for image classification systems, the local features are extracted at dense regular locations [64] [83] [113] [108] [123] [112] [56] [41]. The extracted local features are then aggregated to form global image representation. The existing aggregation methods are similar to the approaches developed for visual search such as BoW, FV, VLAD, which we already reviewed in Chapter 2. Finally, the goal of supervised classification is to learn a function which automatically assigns labels to arbitrary images based on the image representation.
Figure 6.1: Image classification pipeline consists of three steps: (1) Dense feature extraction from images, (2) Aggregation of local descriptors to form global image representation, (3) Learning a classification function which assigns labels to each image.

In this chapter we develop an efficient and effective image classification system based on RVD framework. Our main contributions include:

- We propose a novel method to incorporate second order statistics (diagonal covariance of the residual vectors) into the original RVD and RVD-W frameworks. Our representation gives excellent classification performance even with linear classifiers.

- We present a thorough experimental study to illustrate the effects of various elements and parameters in the RVD-W classification pipeline. For instance, we investigate the impact of dimensionality reduction of local descriptors, size of the visual vocabulary, cluster-wise whitening, normalization of global descriptors and different types of Spatial Pyramid Matching.

This chapter is organized as follows: Section 6.1 provides brief overview of global descriptors used for object classification task, followed by introduction of our extended
6.1 Global Descriptors for classification

This section surveys a number of global image representations targeting classification. We analyse descriptors that encode higher order statistics of the local image features, namely VLAT, higher order VLAD (H-VLAD), and Fisher Vector. We then introduce an effective method to add second order statistics to our original RVD framework.

6.1.1 Extending VLAD to H-VLAD

As introduced in Section 2.3.3, the VLAD descriptor encodes the positions of local descriptors in each voronoi region by computing their residuals with respect to the nearest visual word. A pre-computed codebook \( \{\mu_1, ..., \mu_n\} \) of n cluster centres is used. The residual vectors \( x_t - \mu_j \) are accumulated to obtain cluster-level representations \( \zeta_j \) (Eq. 2.15). The final VLAD representation is obtained by concatenating all aggregated vectors \( \zeta_j \) for all n visual words.

In order to improve the classification performance, Picard et al. [92] introduced VLAT descriptor formed by aggregating tensor products of local descriptors.

In [85], Peng et al. incorporated second order statistics (variance) and third order statistics (skewness) in the VLAD framework. More precisely, the second-order vector is computed using variance of descriptors per cluster:

\[
\zeta_j^c = \frac{1}{N_j} \sum_{x_t: \text{NN}(x_t) = j} (x_t - \varsigma_j)^2 - \sigma_j^2
\]

where \( N_j, \varsigma_j \) and \( \sigma_j^2 \) denote the count, mean and variance of descriptors assigned to cluster \( \mu_j \).
6.1. Global Descriptors for classification

Table 6.1: Performance of VLAD, VLAT and H-VLAD in terms of mean Average Precision (mAP)

<table>
<thead>
<tr>
<th>Method</th>
<th>PASCAL VOC 2007</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAD [116]</td>
<td>54.7</td>
</tr>
<tr>
<td>VLAT [33]</td>
<td>60.8</td>
</tr>
<tr>
<td>H-VLAD [85]</td>
<td>61.2</td>
</tr>
</tbody>
</table>

The third-order vector is computed using skewness which captures the measure of asymmetry of descriptors around their mean vector:

\[
\zeta_j^3 = \frac{1}{N_j} \sum_{x_t: NN(x_t) = j} (x_t - \varsigma_j)^3 \left( \frac{1}{N_j} \sum_{x_t: NN(x_t) = j} (x_t - \varsigma_j)^2 \right)^{-\frac{3}{2}} - \varrho_j
\]  

where \( \varrho_j \) is the skewness of \( j \)-th cluster. The first, second and third order VLAD are concatenated to form the high-dimensional H-VLAD.

Performance comparison of VLAD, VLAT and H-VLAD

Table 6.1 compares the performance of VLAD, VLAT and H-VLAD representations on the PASCAL VOC 2007 benchmark [32]. It can be clearly observed that there is significant increase in the classification performance from 1st-order (VLAD) to the combination of 1st-order and 2nd-order (VLAT) and a further modest improvement of 0.4% when third order is incorporated in H-VLAD.

The improvements to performance resulting from including the second orders motivated us to extend the RVD representation in a similar way.

6.1.2 Fisher Vectors

Fisher Vector representation was introduced in section 2.3.2, it is computed by taking the gradient of the log-likelihood of a set of extracted local descriptors \( X \) with respect to the GMM parameters. The gradients with respect to the mean \( \mu_j \) and the standard
deviation $\sigma_j$ of Gaussian $j$, are denoted by $\zeta_j$ and $\zeta_j^c$:

$$\zeta_j = \frac{1}{T_I \sqrt{\omega_j}} \sum_{t=1}^{T_I} \tau_{tj} \frac{x_t - \mu_j}{\sigma_j} \quad (6.3)$$

$$\zeta_j^c = \frac{1}{T_I \sqrt{2 \omega_j}} \sum_{t=1}^{T_I} \tau_{tj} \left[ \left( \frac{x_t - \mu_j}{\sigma_j} \right)^2 - 1 \right] \quad (6.4)$$

where $\tau_{tj}$ is the soft assignment of descriptor $x_t$ to Gaussian $j$. The FV representation $\zeta_\Theta$ of an image is obtained by concatenating vectors $\zeta_j$ and then of the vectors $\zeta_j^c$ for each of the component Gaussian.

### 6.1.3 Extending the RVD for classification tasks

As described in Section 3.1, the RVD representation $\zeta_j$ of an image is computed by aggregating weighted residual vectors $r_{tj}$ across all neighbourhood ranks.

Since higher order model provides a richer representation and has also shown to improve VLAD, this motivated us to extend our core RVD method with higher order statistics.

The second order RVD is computed as follows:

**Off-line Stage:** Given a set of $N$ residual vectors $((x_1 - \mu_j), (x_2 - \mu_j), ..., (x_N - \mu_j))$ in $\mathbb{R}^d$ extracted from training images, we compute the mean $\varphi_j$ and variance $\sigma_j^2$ of residual vectors for each cluster $j$:

$$\varphi_j = \frac{1}{N_j} \sum_I \sum_{\gamma=1}^K \sum_{x_t:NN(j\gamma)(xt)=j} x_t - \mu_j \quad (6.5)$$

$$\sigma_j^2 = \frac{1}{N_j} \sum_I \sum_{\gamma=1}^K \sum_{x_t:NN(j\gamma)(xt)=j} ((x_t - \mu_j) - \varphi_j)^2 \quad (6.6)$$

**On-line Stage:** Given a query or database image, the second order RVD representation for each cluster $j$ can be formulated as:

$$\zeta_j^c = \sum_{\gamma=1}^K \sum_{x_t:NN(j\gamma)(xt)=j} \tau_{tj} \left[ \frac{((x_t - \mu_j) - \varphi_j)^2}{\sigma_j^2} - 1 \right] \quad (6.7)$$

where $\tau_{tj}$ represents the rank assignment weights.
6.1. Global Descriptors for classification

We apply L2-normalisation to the individual $\zeta_j$ and $\zeta^c_j$ vectors. The RVD representation $R$ of an image is obtained by stacking of the component vectors $\zeta_j$ and then of the vectors $\zeta^c_j$ for each of the $j^{th}$ cluster.

**Higher order RVD-P and RVD-W**

In visual search task, the RVD-P and RVD-W representations achieved significantly enhanced performance, therefore we also expect them to deliver good performance in classification.

In the RVD-P approach, the weighted residual vectors $r_{tj}$ are projected via a local PCA matrix $P_j$ before aggregation into cluster level representation $\zeta_j$ (Eq. 3.7). The RVD-P representation $R^p$ of an image is obtained by concatenating vectors $\zeta_j$ and vectors $\zeta^c_j$ (Eq. 6.7) for each cluster.

In the RVD-W method, the weighted residual vector $r_{tj}$ are transformed via a local PCA matrix $P_j$ and subsequently whitened before aggregation into cluster level representation $\zeta_j$. The $\zeta_j$ and $\zeta^c_j$ vectors are stacked to form the final RVD-W representation $R^w$.

6.1.4 Improving RVD-W representation for object classification

This section improves RVD-W representation by applying power normalization and Spatial Pyramid Matching (SPM). Our discussion is based on RVD-W but these improvements can also be applied to Fisher Vectors and VLAD representations.

**Power normalization**

The RVD-W descriptor undergoes power normalization (PN) [88]. Each component $i$ of vector $R^w$ is transformed using equation 6.8

$$f(y) = \text{sign}(y)|y|^\alpha$$

(6.8)

with $0 \leq \alpha \leq 1$. 
6.1. Global Descriptors for classification

Figure 6.2: Distribution of the coefficients in the first dimension of the RVD-W: (a) with no power normalization, (b) with power normalization ($\alpha = 0.5$).

The application of power norm makes the distribution of the features in a corresponding dimension of $R^w$ less peaky around zero. Figure 6.2 shows that power normalization (with $\alpha = 0.5$) unsparsifies the RVD-W representation and makes it more suitable for similarity comparisons between vectors.

We studied the impact of power normalization factor alpha on the classification accuracy of RVD-W and FV descriptors on PASCAL VOC 2007.

Figure 6.3: Impact of power normalization factor alpha on the classification accuracy of RVD-W and FV descriptors on PASCAL VOC 2007.
accuracy of RVD-W and FV representations. It can be observed from Figure 6.3 that $\alpha=0.5$ provides optimal results on PASCAL VOC dataset. We therefore use $\alpha=0.5$ in all subsequent experiments.

**Spatial Pyramid matching**

In order to capture more detail about the structure of the scene in an image, incorporated FV with in the Spatial Pyramid Matching (SPM). More precisely, the image is split into several sub regions with different granularity and each region is described separately by FV. The FV+SPM representation is formed by stacking of the FVs computed from all aforementioned sub-regions and the FV computed over the whole image.

<table>
<thead>
<tr>
<th>Method</th>
<th>PASCAL VOC 2007</th>
</tr>
</thead>
<tbody>
<tr>
<td>FV</td>
<td>59.9</td>
</tr>
<tr>
<td>FV SPM</td>
<td>62.7</td>
</tr>
<tr>
<td>RVD-W</td>
<td>62.3</td>
</tr>
<tr>
<td>RVD-W SPM</td>
<td>64.8</td>
</tr>
</tbody>
</table>

Table 6.2: RVD-W and FV representation in a spatial pyramid framework (all results in mAP(%) )

It can be seen from Table 6.2 that the SPM significantly improves the classification performance of FV. We also implemented and evaluated SPM framework with the RVD-W representation and found that SPM provides a gain of +2.1%.

**6.2 Experiments**

The purpose of this section is to comprehensively evaluate the effectiveness of our approach on state-of-the-art classification datasets. We first present the experimental setup followed by detailed evaluation and analysis on PASCAL VOC 2007 dataset. We then evaluate the performance of our best pipeline on Caltech256 and MIT scene67 dataset.
Table 6.3: Comparison of SIFT and RootSIFT descriptors (all results in mAP(%) )

<table>
<thead>
<tr>
<th>Method</th>
<th>SIFT</th>
<th>RootSIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>FV</td>
<td>59.0</td>
<td>59.9</td>
</tr>
<tr>
<td>RVD-W</td>
<td>61.3</td>
<td>62.3</td>
</tr>
</tbody>
</table>

Experimental setup

From each image, we extract a set $X = \{x_t, t = 1...T\}$ of dense SIFT descriptors, using a spatial stride of 4 pixels at 9 different scales with $\sqrt{2}$ scale increments (default settings in the VLFeat toolbox [116]). The SIFT descriptors are converted to RootSIFT [7] and projected to 80-dim space using PCA. K-means Clustering is performed on PCA-reduced descriptors to learn a codebook $\{\mu_1, ..., \mu_n\}$ of $n = 256$ cluster centres.

6.2.1 Evaluation on PASCAL VOC 2007

We perform experiments on PASCAL VOC 2007 dataset [32] to optimise our classification pipeline. The PASCAL VOC-2007 dataset consists of about 10k images with twenty different object classes. We follow the standard experimental procedure which comprises of training and validating on the 5011 training images and testing on 4952 test images. The parameters of our method (dictionaries, cluster level PCA and whitening matrix) are learned using the canonical training subset. For each category, a linear one-vs-all SVM classifier [89] is trained using default VLFEAT hyper-parameters ($C=10$ and the number $e$ of epochs is 100) and the performance is measured as mAP over the 20 classes.

SIFT vs RootSIFT

In our first experiment we investigate the benefits provided by the RootSIFT operation. Table 6.3 demonstrates that the conversion improves the classification performance of the RVD-W and FV representations. The SIFT descriptors are transformed to RootSIFT in two steps including L1-normalization on SIFT vectors and square root applied individually to each element.
In previous chapters, we have shown the benefit of SIFT dimensionality reduction, via PCA transform, in the context of visual search. In this section we investigate the impact of dimensionality reduction on classification performance. The pipeline for the experiment is as follows. First, the dimensionality of RootSIFT descriptors is reduced from 128 to \( d' \) dimensions and a codebook of 256 cluster centres is learned. Second, the RVD-W and FV representations are computed and power+L2 normalization is applied. Finally, a linear one-vs-all SVM classifier is trained for each category and the performance is measured as mean Average Precision (mAP) over the all categories. We change the dimensionality of descriptors after PCA (\( d' \)) and observe the changes to classification performance.

The results presented in Figure 6.4(a) show that dimensionality reduction is essential to obtain good classification performance for both representations. For RVD-W, the mAP is 56.5\% without dimensionality reduction, while mAP of 58.8\% is achieved using only 32 most energetic dimensions. It can be observed that optimum performance of 62.3\% is reached when the top 80 dimensions are retained. A similar behaviour is observed for the FV representation: the mAP is only 55.1\% on full SIFT vectors (i.e. without applying PCA transformation), which increases to 59.9\% when the descriptor dimensionality is reduced to 80 dimensions. Compared to FV, RVD-W brings a consistent benefit of about 2\%.
Impact of the codebook size

Another important study investigates the impact of codebook size. Figure 6.4(b) shows that the classification performance increases as we increase the size of the codebook. For $n = 512$, RVD-W and FV obtain mAP of 63.9% and 61.5% respectively. The slope of the curve indicates that further gains could be achieved by increasing the number clusters even further. However for high values of $n$, the dimensionality of the signature becomes prohibitively high. In all the following experiments, the size of the visual vocabulary is fixed to 256, as this value is considered a good trade-off between performance and complexity.

First order global descriptor

We now compare the performance of first order RVD, RVD-P, RVD-W and FV on the PASCAL VOC dataset. The first order FV for each image is computed using VLFEAT toolbox [116]. All global representations undergo power normalization ($\alpha$=0.5) followed by L2-normalization. The dimensionality ($D$) of global representations is thus $80 \times 256 = 20480$. It can be observed from Figure 6.5(a) that all representations based on the RVD framework perform significantly better than FV. Compared to FV, RVD-W offers a significant gain $+2.8\%$ in mAP.
6.2. Experiments

Figure 6.6: Impact of Spatial Pyramid Matching (SPM) on PASCAL VOC dataset: (a) SPM2 with configuration 1×1, 3×1, (b) SPM3 with configuration 1×1, 3×1, 2×2.

Extending RVD with second order statistics

Here we extend the descriptors by adding second order statistics as explained in section 6.1.3. The global representation is computed by concatenating vectors $\zeta_j$ and then vectors $\zeta_j^c$ for each cluster and the global descriptor is power normalized ($\alpha = 0.5$). The dimensionality ($D$) of RVD, RVD-P, RVD-W and FV is equal to $2 \times 80 \times 256 = 40960$. Results are shown in Figure 6.5(b) where it can be observed that second order statistics brings significant gain in classification performance for all global representations. It can also be seen that RVD-W obtains a mAP=62.3% compared to 59.9% for FV, thus offering 2.4% gain.

Spatial Pyramid Matching (SPM)

We discussed in section 6.1.4 that Spatial Pyramid Matching benefits RVD-W and FV by incorporating weak geometric information. In this section we evaluate RVD-W and FV performance using two spatial pyramid configurations SPM2 and SPM3. In the SPM2, an image is partitioned into 1×1, 3×1 sub-regions and corresponding RVD-Ws are computed and concatenated. This creates a descriptor with an overall dimension of $4 \times D = 163840$ elements. In the SPM3, the pyramid divides each image into 1×1, 3×1, 2×2 sub-regions (illustrated in Figure 6.7) resulting in a $8 \times D = 327680$ dimensional
RVD-W vector.

![Spatial Pyramid Matching configuration](image)

Figure 6.7: Spatial Pyramid Matching with configuration $[1 \times 1, 3 \times 1, 2 \times 2]$

Figure 6.6 shows that RVD-W performs significantly better than FV bringing an improvement of 2.1% for SPM2 and improvement of 1.3% for the SPM3 configuration.

A detailed class by class comparison of RVD-W and FV descriptors is presented in Figure 6.8. It can be observed that for majority of classes the best performance is achieved by RVD-W SPM3 method, however it has twice the size of representation used by RVD-W+SPM2.

### 6.2.2 Caltech 256

We now evaluate our systems on the Caltech-256 image classification benchmark. The Caltech-256 is a challenging set of 256 categories containing approximately 30K images. We use the default VLFEAT toolbox settings: the number of training images per category is set to 30. For each category, a linear one-vs-all SVM classifier is trained using default hyper-parameters ($C=10$ and the number $e$ of epochs is 100). The system performance is measured using mean Class Accuracy (mCA) defined as the mean of the diagonal of confusion matrix. We repeated each experiment 10 times and take average of the results to remove uncertainty.

All other parameters remain unchanged, i.e we use dense SIFT with spatial stride of 4 pixels at 9 different scales with $\sqrt{2}$ scale increments. The SIFT descriptors are converted to RootSIFT and their dimensionality is reduced by PCA to 80 dimensions. K-
### 6.2. Experiments

<table>
<thead>
<tr>
<th>CLASSES</th>
<th>FV FIRST</th>
<th>FV FIRST+ SECOND</th>
<th>FV SPM2</th>
<th>FV SPM3</th>
<th>RVD-W FIRST</th>
<th>RVD-W FIRST+SECOND</th>
<th>RVD-W SPM2</th>
<th>RVD-W SPM3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>77.30</td>
<td>80.58</td>
<td>83.05</td>
<td>82.08</td>
<td>77.09</td>
<td>81.72</td>
<td>83.69</td>
<td><strong>83.13</strong></td>
</tr>
<tr>
<td></td>
<td>59.20</td>
<td>64.49</td>
<td>67.41</td>
<td>69.09</td>
<td>64.58</td>
<td>69.14</td>
<td>71.90</td>
<td><strong>72.07</strong></td>
</tr>
<tr>
<td></td>
<td>48.46</td>
<td>56.58</td>
<td>59.35</td>
<td>58.80</td>
<td>52.79</td>
<td>59.44</td>
<td><strong>61.37</strong></td>
<td>60.53</td>
</tr>
<tr>
<td></td>
<td>71.46</td>
<td>72.64</td>
<td>73.68</td>
<td>73.05</td>
<td>73.80</td>
<td>76.58</td>
<td>76.55</td>
<td><strong>76.94</strong></td>
</tr>
<tr>
<td></td>
<td>27.90</td>
<td>32.21</td>
<td>34.49</td>
<td>32.99</td>
<td>27.23</td>
<td>32.44</td>
<td><strong>35.10</strong></td>
<td>34.30</td>
</tr>
<tr>
<td></td>
<td>63.53</td>
<td>64.45</td>
<td>68.72</td>
<td>71.00</td>
<td>63.15</td>
<td>66.78</td>
<td>70.12</td>
<td><strong>71.12</strong></td>
</tr>
<tr>
<td></td>
<td>76.87</td>
<td>78.85</td>
<td>79.95</td>
<td>80.51</td>
<td>77.98</td>
<td>80.59</td>
<td><strong>81.44</strong></td>
<td>80.82</td>
</tr>
<tr>
<td></td>
<td>51.50</td>
<td>57.16</td>
<td>62.98</td>
<td>62.27</td>
<td>58.80</td>
<td>61.62</td>
<td><strong>66.42</strong></td>
<td>65.74</td>
</tr>
<tr>
<td></td>
<td>47.77</td>
<td>51.81</td>
<td>55.94</td>
<td>55.71</td>
<td>50.26</td>
<td>53.70</td>
<td>56.54</td>
<td><strong>57.53</strong></td>
</tr>
<tr>
<td></td>
<td>37.24</td>
<td>43.32</td>
<td>49.97</td>
<td>51.80</td>
<td>47.44</td>
<td>51.27</td>
<td>55.30</td>
<td><strong>55.71</strong></td>
</tr>
<tr>
<td></td>
<td>44.52</td>
<td>53.84</td>
<td>57.08</td>
<td><strong>61.76</strong></td>
<td>50.05</td>
<td>53.87</td>
<td>58.06</td>
<td>60.85</td>
</tr>
<tr>
<td></td>
<td>39.65</td>
<td>45.47</td>
<td>45.62</td>
<td>46.65</td>
<td>41.50</td>
<td>47.40</td>
<td>49.18</td>
<td><strong>50.01</strong></td>
</tr>
<tr>
<td></td>
<td>75.65</td>
<td>78.19</td>
<td>80.05</td>
<td>80.53</td>
<td>75.24</td>
<td>79.71</td>
<td>80.35</td>
<td><strong>80.55</strong></td>
</tr>
<tr>
<td></td>
<td>61.01</td>
<td>65.04</td>
<td>68.29</td>
<td>68.77</td>
<td>65.73</td>
<td>67.57</td>
<td>69.74</td>
<td><strong>70.67</strong></td>
</tr>
<tr>
<td></td>
<td>80.98</td>
<td>83.66</td>
<td>85.13</td>
<td>85.58</td>
<td>80.33</td>
<td>84.05</td>
<td>85.70</td>
<td><strong>86.41</strong></td>
</tr>
<tr>
<td></td>
<td>29.32</td>
<td>32.24</td>
<td>33.75</td>
<td>34.46</td>
<td>30.91</td>
<td>34.24</td>
<td><strong>40.24</strong></td>
<td>37.96</td>
</tr>
<tr>
<td></td>
<td>40.89</td>
<td>44.44</td>
<td>49.16</td>
<td>48.97</td>
<td>46.07</td>
<td>52.40</td>
<td>55.43</td>
<td><strong>55.59</strong></td>
</tr>
<tr>
<td></td>
<td>50.15</td>
<td>55.56</td>
<td>59.33</td>
<td>60.42</td>
<td>51.52</td>
<td>58.75</td>
<td>58.98</td>
<td><strong>61.08</strong></td>
</tr>
<tr>
<td></td>
<td>78.51</td>
<td>81.73</td>
<td>83.04</td>
<td>82.57</td>
<td>78.37</td>
<td>82.27</td>
<td>83.16</td>
<td><strong>83.20</strong></td>
</tr>
<tr>
<td></td>
<td>48.70</td>
<td>56.87</td>
<td>57.64</td>
<td><strong>58.09</strong></td>
<td>46.51</td>
<td>53.18</td>
<td>56.34</td>
<td>56.13</td>
</tr>
</tbody>
</table>

Figure 6.8: Image classification performance on PASCAL VOC dataset. **FV**-Fisher Vectors, **RVD-W**- Robust Visual Descriptor with local Whitening, **FIRST**-first order global descriptor, **FIRST+SECOND**-first+second order global descriptor, **SPM2**-Spatial Pyramid Matching [1×1, 3×1], **SPM3**- Spatial Pyramid Matching [1×1, 3×1, 2×2].
6.2. Experiments

means Clustering is performed on PCA-reduced descriptors to learn a codebook with 256 cluster centres.

First order global descriptor

In Figure 6.9 we compare the first order representations RVD, RVD-P, RVD-W and FV and it can be seen that all RVD representations outperform FV. As expected, RVD-W performs the best achieving improvements in classification accuracy over FV of 2.9%.

First+Second order global descriptor

The global descriptor is formed by concatenating first and second order representations for each cluster. Firstly it can be observed from Figure 6.5(b) that second order statistics brings significant gain in classification performance for all global representations. Secondly, RVD-W signature provides a gain of +2.1% in terms of mean class accuracy, over FV.

Spatial Pyramid Matching

We now evaluate the impact of SPM with two different configurations: SPM2 \([1\times1, 3\times1]\) and SPM3 \([1\times1, 3\times1, 2\times2]\). Compared to FV, the RVD-W representation main-
6.2. Experiments

Figure 6.10: Impact of Spatial Pyramid Matching Caltech256 dataset: (a) SPM2 with configuration $1 \times 1, 3 \times 1$, (b) SPM3 with configuration $1 \times 1, 3 \times 1, 2 \times 2$

Figure 6.11: (a) First order global representations performance, (b) First+Second order global representation performance on MIT Scene 67 dataset.

We now present the results on the MIT Scene 67 [94] which contains about 15620 images of 67 indoor categories. We follow the standard procedure which consists of training on 5630 training images (80 from each class) and testing on 1340 test images (20 from each class). For each class, a linear one-vs-all SVM classifier is trained and the performance is measured using mean Class Accuracy (mCA). Figure 6.11(a)
6.3 Comparison with the state of the art

The upper section of Table 6.4 lists the performance of global image representations derived from shallow local descriptors. It can be seen that the proposed RVD-W outperforms all prior-art shallow representations, in particular it improves dramatically (gain of +10% mAP), over VLAD on both PASCAL VOC 2007 and MIT Scene 67 datasets. It can be also be observed that RVD-W provides a significant boost of +8.8%, +7.7% and +6.8% in mAP (PASCAL VOC 2007) when compared to VQ, LLC and SV. The
Table 6.4: Performance of global representations on state-of-the-art datasets, VQ: Vector Quantization, LLC: Linear Local Coding, SV: Supervectors, FV: Fisher Vectors and RVD-W.

<table>
<thead>
<tr>
<th>Method</th>
<th>Vocabulary size</th>
<th>PASCAL VOC 2007</th>
<th>Caltech 256</th>
<th>MIT Scene 67</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAD [116]</td>
<td>256</td>
<td>54.7</td>
<td>-</td>
<td>53.3</td>
</tr>
<tr>
<td>VLAT [93]</td>
<td>512</td>
<td>60.8</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>H-VLAD [85]</td>
<td>256</td>
<td>61.2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VQ [21]</td>
<td>25k</td>
<td>56.1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>LLC [21]</td>
<td>25k</td>
<td>57.3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>SV [21]</td>
<td>1024</td>
<td>58.2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FV [22]</td>
<td>256</td>
<td>63.7</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FV_{our}</td>
<td>256</td>
<td>63.7</td>
<td>48.1</td>
<td>64.9</td>
</tr>
<tr>
<td>RVD-W</td>
<td>256</td>
<td>65.0</td>
<td>49.3</td>
<td>65.4</td>
</tr>
<tr>
<td>CNN [22]</td>
<td>-</td>
<td>79.7</td>
<td>77.6</td>
<td>-</td>
</tr>
<tr>
<td>CNN [101]</td>
<td>-</td>
<td>79.0</td>
<td>74.2</td>
<td>-</td>
</tr>
<tr>
<td>CNN [95]</td>
<td>-</td>
<td>77.2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CNN [118]</td>
<td>-</td>
<td>81.5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CNN [38]</td>
<td>-</td>
<td>80.1</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

RVD-W representation also outperforms FV on all three datasets.

The lower part of the Table 6.4 lists the performance of Convolutional Neural Network (CNN) based representations. These approaches significantly outperforms the shallow encodings on state-of-the-art benchmarks. However, the main focus of the thesis is to aggregate hand-crafted shallow local descriptors into a robust global descriptor.

### 6.4 Conclusion

In this chapter we extended the core RVD-W representation to classification tasks by improving the discriminative power of the global descriptors (RVD, RVD-P and RVD-
W). This was achieved by incorporating second order statistics (variance) in the original framework. Furthermore, the global representation are integrated with Spatial Pyramid Matching scheme to capture weak geometrical information. A detailed evaluation on de-facto standard benchmarks demonstrates that our approach achieves excellent results compared to VLAD and FV.
Chapter 7

Conclusions

In this chapter we summarize our contributions and briefly discuss possible directions for future work.

7.1 Summary of Contributions

Searching content among millions of images is a massively challenging yet important task. The early solutions based on matching local descriptors are computationally expensive. In this thesis we designed, implemented and evaluated a set of robust and discriminative global signatures suitable for web-scale visual search and classification tasks.

We developed a core Robust Visual Descriptor (RVD) which combines rank-based multi assignment with a robust aggregation framework. In our approach, local descriptor are assigned to multiple cluster centres with rank weights, leading to a stable and discriminative global image representation. The residual vectors between the descriptors and their corresponding cluster centres are computed and L1-Normalized. This direction preserving mechanism limits the impact of outliers and performs such that the influence of a single descriptor on the aggregated representative value is similar for all descriptors. The normalized residual vectors are aggregated with rank based assignment weights to yield the core RVD representation, which significantly outperforms existing state-of-the-art global representations on all standard datasets.
We further improved the discriminative power of the core RVD representation by introducing two novel components. The first one de-correlates the residual vectors inside each cluster centre using a local PCA basis before aggregation (RVD-P). In the second approach the weighted residual vectors are whitened in each cluster before aggregation into RVD-W, leading to a balanced energy distribution in each dimension. We also proposed a new post-PCA normalization which improves separability between the matching and non-matching vectors. This new normalization benefits not only our RVD-W descriptor but also improved existing approaches based on FV and VLAD aggregation. Finally, to work with very large databases we encoded RVD-W using the optimised Product Quantization approach. A detailed evaluation demonstrated that RVD-W pipeline achieved very significant gains of +10% in mAP over the most advanced version of VLAD and +16% over Fisher Vectors, on both the Holidays and Oxford datasets. On the large scale datasets Holidays1M and Holidays10M, our method obtains a mAP of 45.1% and 40.5%, while on Oxford1M and Oxford 10M we reach mAP of 35.1% and 30.5%, all significantly outperforming any results published up to date.

To work with web-scale databases, the high dimensional and floating-point RVD-W signatures are converted to binary codes. We also presented a novel descriptor matching algorithm PCAE+WH, where the weights assigned to Hamming distances are proportional to the variance of the projected data in the corresponding dimensions. Our approach improved the retrieval performance by minimizing the quantization error introduced by mapping of the floating-point data to the vertices of the binary Hamming cube. A detailed evaluation on de-facto standard benchmarks demonstrated that our scheme outperforms all state-of-the art methods by a large margin of between 3% and 8%. This performance margin is maintained for large databases, proving the effectiveness of the method.

In the context of industry work on CDVS, we developed an effective and efficient Mobile Visual Search system based on the scalable RVD representation. The scalable RVD signature is obtained by employing Cluster Selection and Bit Selection methods. Extensive experiments on the MPEG CDVS dataset demonstrate excellent retrieval and pairwise matching performance, outperforming the CDVS reference model based on the Scalable Compressed Fisher Vectors.
Finally, we developed an effective image classification system based on the RVD representation. After analysing the performance of higher order VLAD (H-VLAD) and Fisher Vectors, we proposed a method to incorporate second order statistics, represented by the diagonal covariance of the residual vectors, in the original RVD and RVD-W frameworks. The empirical comparisons on challenging benchmarks (PASCAL VOC2007, Caltech256 and MIT Scene67) showed the advantage of RVD and RVD-W when compared to latest techniques such as Fisher Vectors and VLAD.

7.2 Future Work

The focus of this research was to develop a robust and compact global image descriptor for image retrieval and classification. Even though in our work we achieved the best results published to date, still a lot of work needs to be done to achieve human-level success rates in large scale retrieval and classification. Moreover, as the datasets continue growing, these tasks will become even more challenging. In this section, we discuss several promising directions for future work that might lead to better solutions to the problem at hand.

Aggregation of binary and region-based descriptors: Many contemporary pipelines for object recognition and retrieval choose to employ local binary descriptors in order to reduce extraction and matching complexity. This is particularly important in mobile applications, where at least some processing is performed on a terminal with limited resources. Even when server resources are available, computational complexity is still an issue due to the ever increasing scale of databases, image resolutions and the required accuracy and speed of search. Consequently, local binary descriptors become increasingly popular, as they deliver high matching speed, small memory footprint and are relatively fast to extract. While many techniques exist for extracting global representations from floating-point local descriptors, such as SIFT, surprisingly hardly any research exists on how to efficiently aggregate local binary descriptors. Binary descriptors such as BRIGHT [42], FREAK [84] and BRISK [63] are significantly faster to compute compared to SIFT and even SURF, while providing comparable performance. Thus further research on the aggregation of binary descriptors is required.
In this thesis we show that global image representations based on local gradient descriptors perform reasonably well in visual recognition of textured objects, such as buildings, book covers, Dvds and paintings. However, for smooth (fairly texture-less) objects, local descriptors based on the colour and shape properties of an object can be more discriminative. One future direction could be to develop a global signature based on colour and shape descriptors for the recognition of smooth objects.

**Deep learning:** The classification pipeline developed in Chapter 5 is based on shallow representations such as FV, RVD and RVD-W. Recently, several deep representations based on Convolutional Neural Networks (CNN) [15]; [61]; [59] have been introduced for classification tasks. While deep representations achieve high classification performance, FV and RVD classifiers are significantly less costly to train and evaluate. Recently, Perronnin et al. [87] developed a hybrid system that aggregates FV with CNN for large scale image classification- the first unsupervised layers are based on FV while the subsequent supervised layers are based on CNN. It would be interesting to further pursue this direction and implement an architecture that combines RVD-W with CNN.

**Applications based on RVD-W descriptors:** Detection and classification of objects in massive amounts of video is required for a broad range of applications, including surveillance and security, automotive or Digital Asset Management (DAM) systems. We already started development of a system based on RVD-W global descriptors for recognition of objects in videos.

We would be interested in developing a system, which allows a clinician or medical scientist to perform structured image retrieval in large medical image repositories.
Bibliography


