Novel Image Representations for Visual Categorisation with Bag-of-Words

Piotr Koniusz

Submitted for the Degree of
Doctor of Philosophy
from the
University of Surrey

Centre for Vision, Speech and Signal Processing
Faculty of Engineering and Physical Sciences
University of Surrey
Guildford, Surrey GU2 7XH, U.K.

March 2013

Summary

Visual Category Recognition aims at fast classification of objects, as well as scenery, action, and semantically complex concepts in collections of unannotated images. Its applications include security and crime prevention, rapid selection of content for efficient media practices, television and press archives, organisation of visual content in the social media, e-commerce, robotic recognition, and many more.

There exist a variety of approaches to visual categorisation. However, due to complex nature of visual appearances and complex taxonomy of objects, a simplifying statistical model developed for natural language processing, called Bag-of-Words, is typically used.

In such a model, descriptors are extracted from images at keypoint locations and then expressed as vectors representing visual word appearances, referred to as mid-level features. A pooling step is carried out to transform mid-level features from an image into a final vectorial representation called image signature. Finally, a classifier is applied.

Segmentation-based interest points for matching and recognition are investigated first. Two simple methods for extracting features from the segmentation maps are proposed. They focus on the boundaries and centres of the gravity of the segments.

Segmentation-based image descriptors are proposed next. They are extracted from pairs of adjacent regions from an unsupervised segmentation. Thus, semi-local structural appearances are exploited. This limits contribution of uniform regions.

A highly popular technique for coding the local image descriptors in Bag-of-Words, called Soft Assignment, is combined with Linear Coordinate Coding to minimise its quantisation loss which strongly correlates with the best classification performance.

An approach that introduces spatial information to Bag-of-Words, called Spatial Coordinate Coding is proposed. It reduces the size of mid-level features tenfold. Moreover, as dominant orientations of edges and colour are sources of bias in images, we learn them at multiple levels of coarseness by Dominant Angle and Colour Pyramid Matching.

A number of techniques for generating mid-level features as well as various pooling methods that aggregate mid-level features into image signatures are investigated. We generalise these pooling methods to account for the descriptor interdependence and introduce an improved pooling that addresses noise effects in mid-level features.

Bag-of-Words typically extract the first-order statistics from mid-level features. To improve recognition, aggregation over co-occurrences of visual words in mid-level features is proposed. An appropriate derivation is provided and various likelihood inspired pooling operators investigated. Moreover, an extension to multiple modalities is proposed.

Key words: Bag-of-Words, Keypoints, Descriptors, Soft Assignment, Sparse Coding, Spatial Coordinate Coding, Max-pooling, Dominant Angle Pyramid Matching, Mid-level Features, @n Pooling, Tensor, Second-order Occurrence Pooling, Co-occurrences.

e-mail: p.koniusz@surrey.ac.uk
www: http://claret.wikidot.com

Acknowledgements

First, I would like to thank my supervisor, Krystian Mikolajczyk, for providing an informative and apt guidance throughout this research journey, and wish to express my sincere gratitude for all his hard work, patience, stimulating discussions, as well as encouragement. Thanks are also due to members of Centre for Vision, Speech and Signal processing at the University of Surrey. In particular: Mark Barnard for many stimulating discussions and a friendly atmosphere in the shard office, Tim Sheerman-Chase and David Windridge for proofreading a journal draft, Kevin Wells for helping out with the project drafts, and in particular to Fei Yan, whose advice and feedback over the last two years has been invaluable. Moreover, thanks are also due to Denise Bland, Teo De Campos, and Philippe-Henri Gosselin for several insightful discussions.

Many thanks also go to our senior software support officer Bevis King, and Dave Muno, who, on numerous occasions, supplied useful advice and assisted in solving various IT problems, often outside of the regular working hours. Kind acknowledgements are due to John Collomosse, whose little agile servers often provided additional computational power for my research. Sincere thanks are due to Hongping Cai, Zdenek Kalal, and Muhammad Awais for their friendship and a kind office atmosphere, as well as to James Field, our former centre administrator, for his eagerness in solving various administrative conundrums. Many thanks to my friends in Department of Economics for organising motivational coffee breaks and relaxing meals out.

Finally, I would like to thank my family for their emotional support and constant interest in the latest developments in my work. Special thanks are also due to Veronica for her enormous patience, understanding and help in the difficult moments.

I gratefully acknowledge the financial support of the BBC Future Media and Technology and the EPSRC who funded this research under grant EP/F003420/1.

Piotr Koniusz,
February 2013.

Contents

List of Figures

List of Tables

Acronyms and Symbols

Acronyms

@n Pooling of the Top n Largest Coefficients 83

Avg Average pooling 83

AxMin Approximation of MaxExp 83

BoW Bag-of-Words 13

CCTV Close Circuit Television 4

CoPM Colour Pyramid Matching 27

DA Dominant Angle 70

DoPM Dominant Angle Pyramid Matching 27

EC Exact Complementarity 46

EGO Efficient Graph-Based Image Segmentation 32

ExaPro *at least one particular visual word being present in an image* 83

FHNNS Fast Hierarchical Nearest Neighbour Search 94

FK Fisher Vector Encoding 122

Gamma Gamma Correction 83

GMM Gaussian Mixture Model 26

HA Hard Assignment 60

HE Hessian 43

KDA Kernel Fisher Discriminant Analysis 18

LCC Linear Coordinate Coding 26

LcSA Approximate Locality-constrained Soft Assignment 27

LDA Linear Discriminant Analysis 18

LLC Locality-constrained Linear Coding 27

lp-norm ℓ_p norm 83

MAP Mean Average Precision 20

Max Max-pooling 83

Symbols

ε_o	The region overlap
ε_n	The NN overlap
H	A homography matrix relating two images
R_{μ_r}	The reference region
$R_{H^T \mu_p H}$	The projected region
χ^2	The χ^2 distance
χ^2_{RBF}	The χ^2 distance combined with the RBF kernel
θ	Parameters of Gaussian Mixture Model
w_k	Weight of a k^{th} component of GMM
$\boldsymbol{m}$	A vector containing visual word
$\boldsymbol{m}_k$	A visual word with index k Also, a mean of a k^{th} component of GMM
$\mathcal{M}$	A set of visual vocabulary atoms
$\boldsymbol{\sigma}^2$	Covariance of a k^{th} component of GMM
σ^2	Variance of a k^{th} component of GMM
σ	Standard deviation of a k^{th} component of GMM Also, a smoothing factor of SA and LcSA
$\boldsymbol{x}$	A descriptor descriptor vectors
$\boldsymbol{x}_n$	A descriptor descriptor vectors with index n
$\mathcal{X}$	A set of descriptor vectors
D	Descriptor dimensionality
N	Number of descriptor vectors
K	Number of visual vocabulary atoms
G	Gaussian function
$\Lambda(\mathcal{X}; \theta)$	The GMM cost
$p(k\|\boldsymbol{x})$	The membership probability of selecting k given $\boldsymbol{x}$
$f_{\boldsymbol{m}}(x)$	Descriptor mapping to visual word $\boldsymbol{m}$
ξ^2	Quantisation loss for a batch of descriptors
$\xi^2(\boldsymbol{x})$	Quantisation loss for descriptor $\boldsymbol{x}$
c^x	Spatial position x of a descriptor
c^y	Spatial position y of a descriptor
w	Image width
h	Image height
ω	A trade-off between the visual appearance and the spatial bias
$S(Q^l)$	A number of SPM partitions with depth Q and l dimensions
Q	Also, a number of spatial partitions
$\boldsymbol{\phi}$	A mid-level feature
$\boldsymbol{\phi}_n$	A mid-level feature with index n
ϕ_k	A k^{th} coefficient in a mid-level feature
ϕ_{kn}	A k^{th} coefficient in a mid-level feature with index n
$\mathcal{I}$	A set of image indexes
$\mathcal{N}^{\rangle}$	A set of indexes of descriptors of image i
$\mathcal{N}$	A set of indexes of descriptors (of an image in considerations)
$\mathcal{N}_q$	A set of indexes of descriptors falling into partition q
f	A feature coder
g	A pooling operator
$\boldsymbol{h}$	A normalised image signature
$\hat{\boldsymbol{h}}$	A not normalised image signature

$\boldsymbol{\psi}_q$	Pooled feature vector given partition q
$\boldsymbol{\psi}$	Pooled feature vector (given partition)
ψ_{kq}	A k^{th} coefficient in a pooled feature vector given partition q
ψ_k	A k^{th} coefficient in a pooled feature vector (given partition) Also, intermediate statistics for FK and VLAT Also, a vector after flattening a tensor
Ker_{ij}	A value of kernel function between images i and j
$\max(.)$	Maximum between elements of a set
$\text{avg}(.)$	Average between elements of a set
$\text{sgn}(.)$	A sign of a given value
$\text{srt}(., @n)$.	A partial sort for $@n$ largest values
$\text{avg}\,\text{srt}(., @n)$	An average over partial sort for $@n$ largest values
$\mathcal{M}(\boldsymbol{x}, l)$ ...	The l-nearest anchors of descriptor $\boldsymbol{x}$ in vocabulary set $\mathcal{M}$
$p(k\|\boldsymbol{x}, \sigma, l)$	The membership probability of selecting a k^{th} given $\boldsymbol{x}$, σ, and l
$\mathcal{O}(.)$	The big $\mathcal{O}$ complexity notation
ℓ	A dilation of cluster for the FHNNS search
l	The l-nearest neighbours for LcSA and LLC
$\bar{N}$	A correction parameter for MaxExp pooling
γ	A correction parameter for Gamma pooling
β	A correction parameter for AxMin pooling
β or $\beta^{(q)}$ ..	Weights for fusing multiple modalities
$@n$	A number of the top $@n$ mid-level coefficients for the $@n$ scheme
ℓ_p	The ℓ_p norm
ρ	$1/\rho$ denotes radius for the RBF function
$\mathbf{1}$	A vector filled with coefficients equal 1
α	A regularisation parameter for SC and LLC
$\boldsymbol{C}_k$	Covariance of cluster k
$\otimes_r(.)$	A tensor product of order r
r	Order r of occurrence pooling
$u_:$	A tensor flattening operator (upper simplex+diagonal preserved)
$u_:^*$	A tensor flattening operator (all coefficients are preserved)
$ker\left(\boldsymbol{\phi}, \bar{\boldsymbol{\phi}}\right)$	Minor kernel function between two image signatures
$\langle . \rangle$	A dot product
Φ	A set of mid-level features
$\boldsymbol{x}^q$	A descriptor from modality q
$\mathcal{M}^q$	A dictionary for modality q
$\boldsymbol{\phi}^q$	A mid-level feature for modality q
$\oplus_{t=1}^{\overline{T}}(.)$	Concatenation operator over $t=1, ..., \overline{T}$ pyramid levels
$\overline{T}$	Pyramid levels for the special case SPM
$\boldsymbol{Z}$ and $\overline{\boldsymbol{Z}}$ ..	Vector defining splits of the special case SPM
$\boldsymbol{\xi}$	Residual descriptor (vector)

Measures

EC ..	Exact Complementarity	RC	Relaxed Complementarity
MAP	Mean Average Precision	Accuracy	Mean Average Accuracy

Chapter 1

Introduction

The cognition of visual reality can be attributed to the primates and other animals. Perception of visual stimuli is so valuable in the natural habitat that complex image-forming eyes are said to have evolved some 50 to 100 times [Haszprunar, 1999]. Human interactions with objects, simple daily routines, as well as skilled tasks rely on our cognitive abilities to distinguish from $30K$ of objects [Biederman, 1987] according to their utility. Arguably, one of the biggest challenges in Computer Vision is to discover mathematical models that could enhance computers with such an ability to perceive and infer on a par with the human. However, the complexity of visual stimuli and the subtle object taxonomy [Torralba et al., 2008] prove this a formidable task. The Computer Vision community has been focusing on a number of tractable aspects of Visual Category Recognition (VCR):

- *Visual Object Category Recognition* aims at classification of multiple objects of varied nature in collections of unannotated images. The examples of objects include *human*, *cat*, *chair*, *train*, *bottle*, *etc.* A recognition algorithm has to predict which of these objects are present in any given image [Everingham et al., 2007].

- *Scene Category Recognition* extends the above problem to recognition of the environments. The categories of interest often include (but are not limited to) *office space*, *shopping areas*, *kitchen*, *campus*, *forest*, *city*, *country side*, *etc.*

- *Action Recognition* and *Human Action Recognition* focus on determining which activities are performed in any given image [Everingham et al., 2010] (or a video footage). Often, the main goal is to determine the human activity that may either relate to human dynamics (*e.g. running*, *walking*, *boxing*) or human interaction with objects (*e.g. phoning*, *driving*, *reading*).

- *Visual Concept Detection* can be seen as a generalisation of the above problems [ImageCLEF, 2011, Nowak et al., 2011]. It addresses recognition of concepts of a varied nature, including semantically complex topics, *e.g. party life*, *funny*, *work*, *birthday party*, *beautiful*, *violent*, *sport event*, *conference*, *etc.*

Other problems in Computer Vision that are related to VCR also include:

- *Visual Object Detection* and Person Layout Detection that are concerned with locating objects of interest within images, recognising their categories, and delineating them with bounding boxes. For the latter problem, human body parts have to be recognised and delineated, *e.g. head*, *hands*, *arms*, *legs*.

- *Visual Object Segmentation* that determines location of objects of interest, recognises their categories, and provides pixel-wise delineation of their extent in images.

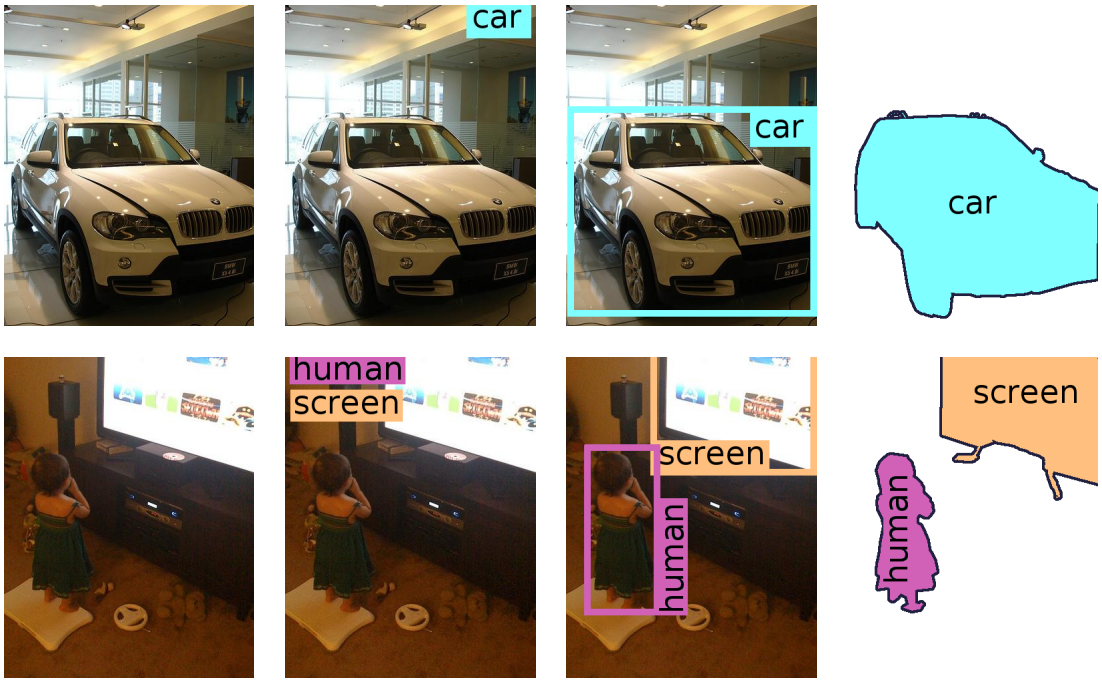
- *Image Retrieval* that addresses fast searching through vast collections of images for the content visually similar to a query image.

- *Medical Recognition* that provides sophisticated warning systems for a variety of medical conditions, *e.g.* recognition of cancer.

- *Remote Sensing* that exploits multispectral image classification, *e.g.* recognition of suspicious buildings during the reconnaissance flights.

- *Face Detection and Recognition* that are concerned with distinguishing faces from backgrounds and recognising subject's identity, respectively.

- *Emotion Recognition* that classifies the body language and facial expressions, *e.g. happy*, *sad*, *angry*, *confusedscared*, *etc.*



(*a*) image (*b*) recognition (*c*) detection (*d*) segmentation

Figure 1.1: Illustration of fundamental problems that Visual Category Recognition deals with. (*a*) An input image, (*b*) Visual Object Category Recognition results in a list of objects in the image, (*c*) Visual Object Detection results in a delineation of these objects, (*d*) Visual Object Segmentation results in the pixel-wise outlines.

- *Gesture Recognition* that aims at classification of the signs and signals expressed by the human body language (often hands and head gestures).

- *Gait Analysis* which is a study of human motion that helps in recognition of subjects' identities. This is a particular example of even wider *Biometric Recognition*.

The above topics do not constitute by any means an exhaustive list of all current directions of research. However, such well-defined topics help study a wide scope of VCR and other related problems as the standardised frameworks for comparison and benchmarking are available. Figure 1.1 illustrates Visual Object Category Recognition, as well as Visual Object Detection and Segmentation. For simplicity, these problems are referred to as classification (or visual categorisation), detection, and segmentation.

This thesis is mainly concerned with Visual Object Category Recognition and Concept Detection in the large collections of images. The methods that will be proposed are also applicable to Human Action Recognition and Scene Category Recognition. They all are referred to as classification or visual categorisation. The datasets, evaluation protocols, prior knowledge in this area, as well as pipelines used for VCR will be described later. The next section motivates this research followed by the background to VCR, typically encountered challenges, and the list of the contributions made in this thesis.

1.1 Motivation

Beside a desire to reconcile the gap between the cognitive capabilities of humans and machines, the VCR systems are applicable in principle in many every day scenarios. In the Digital Economy of the future it is expected that large repositories of digital information of various types will be compiled, stored, and processed for the benefit of people. This includes images, video, sound, and text information. These modalities will require an advanced storage and search technology commonly referred to as Content-based Multimedia Information Retrieval. Currently, YouTube uploads an equivalent of 240000 full-length films every week, over 3 billion videos are viewed daily. Flickr was hosting around 4 billion images at the beginning of 2010. 7 billion pieces of content is shared on Facebook weekly. Nowadays, 85% of the UK population and 30.4% worldwide have instant access to the Internet amounting to staggering 2.1 billion users. It is forecast that the world population will increase from 7 to 9 billion people by 2020 putting strain on both multimedia based economy and security. Therefore, one can envisage numerous applications that employ VCR:

- *Security and Crime Prevention.* Automated recognition of criminal content and challenging behaviours on image and video sharing web services can improve their efficiency and raise high standards of responsible broadcasting of personalised content. Moreover, an automated abnormal and suspicious event and action detection for CCTV appears as a desired pivotal piece of functionality. This not only would reduce operating costs but could help navigate a security officer directly to suspicious behaviours as decided by a well-trained classification algorithm. It would

mitigate social unrest regarding CCTV related privacy breaches. An illustration of the scope for such applications can be found in [INDECT, 2009].

- *Efficient Media Practices.* The usage of Visual Object, Scene, Action, and Concept Recognition for in the media practices can be aptly illustrated by a project in Classification and Retrieval of Images II [Koniusz et al., 2009]:

  > The BBC's News Interactive's User Generated Hub receives hundreds of images per week from the public, though any major incident very quickly increases the number of images received to an unworkable amount. During the London bombings of July 7th hundreds of images were received in a very short space of time. The first pictures of the incident on the BBC's web site were from the public. Such a material is often topical and must be dealt with quickly. This project addresses object recognition and retrieval of images to allow rapid selections to be made.

- *Television and Press Archives.* Public access to the vast television and press archives can be also enhanced by VCR. For instance, the BBC has the largest audio-visual archive in the world that is planned to be opened up for on-line public access [BBC Press Office, 2008]. The BBC's actions are part of a much larger initiative for cultural institutions to release large sections of their material. However, there exist technical challenges. The audio-visual content has often a very basic description, *e.g. title*, *transmission date*, *synopsis*, *genre*, and *contributors*. Hence, there is a need to enhance the ways of discovering content through automated audio-visual searches as opposed to traditional text based approaches.

- *E-commerce Engines.* The customer on-line shopping experience can be enhanced by applying the retrieval techniques that let users take photos of items and browse for close matches amongst the items on sale. Moreover, sellers could gather all details about the items they are about to sell with a single photo query that is then matched against a dataset of commercial goods. Such facilities may be particularly of use when textual annotation is ambiguous or scarce. A changing face of car sales provides an interesting case study [Jung, 2012].

- *Social Media Networking.* Large user generated photo collections are available on Flicker, Picassa, and other web services. Ability to organise these collections by categories of objects, concepts, genre, and moods could enhance on-line public access to the photographs on the computing clouds [Huiskes and Lew, 2008].

- *Robotics and Planetary Explorations.* A desire to have autonomous robots that function in a complex environment means these machines have to recognise a variety of objects, scenes, and other environmental and geological features. There is an increasing trend of using ever more autonomous exploratory vehicles in environments inherently hostile to humans, *e.g.* Mars Rover or Mars Express exploring Mars. These vehicles could perform an autonomous visual analysis of obstacles to avoid. Moreover, the public was recently asked to help classify various geological features in over 3 millions of images of the Martian surface taken by the Mars Reconnaissance Orbiter [Zooniverse, 2012]. Having such amount of labelled data could enable training and autonomous detection of unusual features.

- *Medical Diagnosis and Well-being.* There is an ongoing effort in development of the medical search and classification engines. Diagnostic images from radiology, dermatology, microscopy, as well as complex tomography and magnetic resonance can be used in training and classification for potential health hazards [Tommasi and Deselaers, 2010, Mller and Kalpathy-Cramer, 2010]. Moreover, the collapse detection systems are hoped to improve quality of elderly patients' life.

- *Monitoring Wildlife Populations.* The accurate estimation of wildlife population density is difficult and requires considerable investment of resources and time. Amongst many tools, the status of a wildlife population can be monitored in some cases by usage of either satellite and aerial photography or even land infrared thermal imaging stations. As the biology and ecology of the species of interest vary, this topic poses constant new challenges [Witmer, 2005].

- *Well-being of Animals in Research.* Balancing animal-based research with animal well-being is of great relevance [Weed and Raber, 2005]. A well-being of the animals used in support of the research is often under the public scrutiny. The VCR systems could provide a solution to non-invasive monitoring of the quality

of animal sleep. The goal could be to distinguish between periods of comfort and distress in order to bring some quality into sleep.

- *Industrial and Food Quality Control.* Due to a variety of industrial and food products and ever changing regulations, there is a constant need for bespoke visual inspection. This subject is widely studied, yet it always faces new challenges.

To facilitate applicability of VCR for the above problems, one has to address shortcomings of the state-of-the-art classification systems. Arguably, a long term goal is to achieve accuracy closer to the human cognitive skills and improve their time complexity. A simplifying statistical model developed for natural language processing, called Bag-of-Words [Sivic and Zisserman, 2003, Csurka et al., 2004], is often used to address challenges such as complex nature of visual appearances and difficult taxonomy of objects. The basic variants of such a model are explained in section 1.2. Bag-of-Words is comprised of several functional modules, each having a strong impact on the quality of image representation. Moreover, the interaction between these modules has to be taken into account to assure that outputs of one unit match inputs of the next unit. Historically, improving visual categorisation relied on capturing a variety of complementary modalities from images [Nilsback and Zisserman, 2006, Bosch et al., 2007, Tahir et al., 2010]. We note that the edges of objects, entire object regions, textures, and numerous colour spaces can be utilised together. Recent improvements applied to Bag-of-Words highlighted that it is also possible to attain state-of-the-art visual categorisation with a single modality rather than multiple cues [Yang et al., 2009]. Therefore, our technical motivation is to study each of the modules in Bag-of-Words independently, propose improvements based on a number of identified shortcomings, and also consider an interplay between these modules. The list of contributions made in this thesis is provided in section 1.5 while below are the details of technical motivation:

- In Bag-of-Words, multiple local image appearances are captured from an image at keypoint locations that indicate visually rich regions of interest. Such features are biologically inspired, however, they remain to be handcrafted. This provides the scope to further investigate how to capture objects in images robustly, what constitutes good features, and how to detect informative regions of interest.

- These features are next expressed as vectors representing visual word appearances. This constitutes an analogy to Bag-of-Words for natural language processing. Due to the differences between the visual reality and text, this thesis seeks to understand what is the exact role of this step and how to best perform it.

- Lastly, multiple visual word appearances are typically aggregated into a final vectorial representation which enables training and classification. We seek to understand how to best aggregate the input information in this step, what evidence has to be retained and why, and how the previous step affects this procedure.

1.2 Background

Visual Object Category Recognition and Concept Detection often employ three tasks:

- extraction of low level features from images, as outlined in section 1.2.1

- transformation of these features into succinct image representations that can be compared against each other, as explained in section 1.2.2

- classification performed on these representations, as presented in section 1.2.3

1.2.1 Feature Extraction

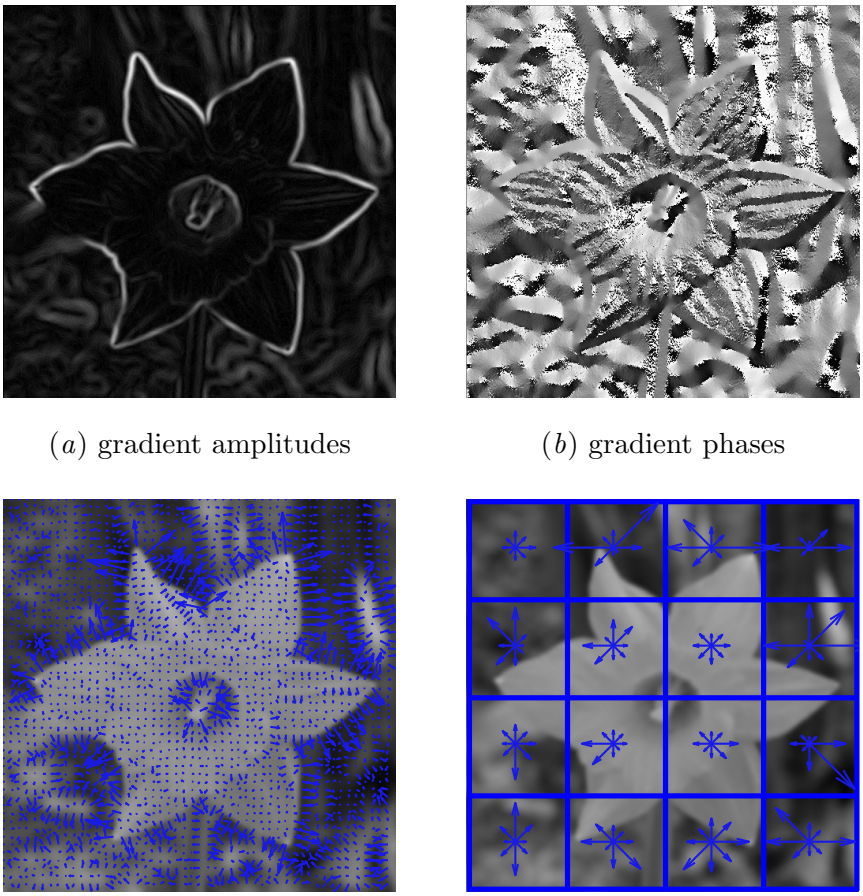
For this step, various global and local descriptors have been proposed to date. In Scene Category Recognition, *global image descriptors* are often used as scenery tends to dominate an entire image. Therefore, the global appearance is relevant in such a recognition problem. However, Visual Object Category Recognition has to deal with objects that appear at various scales and orientations in images. *Local image descriptors* are often employed for such a task. They are typically characterised by the degree of their invariance to geometric and photometric image transformations [Mikolajczyk and Schmid, 2005]. Moreover, as such local image descriptors operate on image patches, they require a strategy for sampling these patches from an image. Often, *interest point detectors* that determine blob and corner structures in images at multiple spatial scales

are employed. The descriptors are centred on such keypoints and the surrounding content is then described. The quality of these detectors is determined by measuring *the repeatability* of discovered interest points under common image transformations [Mikolajczyk et al., 2005]. Another popular strategy in VCR, where descriptors are extracted at predefined spatial intervals for predefined number of spatial scales, is called *dense sampling*. [Nowak et al., 2006].

Global Image Descriptors. Such descriptors are aimed at the efficient analysis of global scenes. Local image statistics are extracted across several spatial regions a.k.a. *spatial girds* that cover an entire image. These statistics may compete locally for the winner-takes-all to represent a given local region. They are then concatenated into holistic image representations. For example, set of biologically inspired early-visual features are extracted in [Siagian and Itti, 2007] by computing statistics at multiple spatial scales in so-called feature channels to account for colour, intensity, orientation, flicker and motion. These operations are repeated in every spatial region.

Local Image Descriptors. They transform image patches into local image representations that remain stable (to a certain degree) under image rotation, scale and viewpoint changes, small translation, varied brightness, blur, and inconsistency of colour. Such descriptors have to match similar objects under these transformations and yet separate appearances for different classes of objects. A Scale Invariant Feature Transform (SIFT) descriptor [Lowe, 1999] fulfils the above needs and is widely used in VCR.

The operating principle of this descriptor can be explained in the following steps: i) the image gradients are computed for every pixel on the luminance channel by convolving a given patch with the vertical and horizontal operators $[-1\,0\,1]$ and $[-1\,0\,1]^T$, ii) these gradients are transformed into two matrices of the gradient amplitudes and phases, respectively, iii) for every pixel, the gradient phase and spatial location within the patch are typically quantised into one of 8 angular and 4×4 vertical and horizontal values, iv) for every pixel, such a quantized value determines which vector bin (one from $8\times4\times4$) is updated by the corresponding gradient amplitude. The final vector is then ℓ_2 norm normalised. Often, additional steps are performed: v) the gradient amplitude is weighted by a Gaussian window imposed over the patch, vi) bilinear interpolation



(*a*) gradient amplitudes (*b*) gradient phases

(*c*) amplitudes and phases as vectors (*d*) final descriptor

Figure 1.2: Steps required to compute the SIFT descriptor. (*a*) The map of gradient amplitudes is computed from the vertical and horizontal maps of gradients. (*b*) The corresponding map of gradient phases. (*c*) The amplitude and phase are illustrated as vectors (note they are orthogonal to the boundaries of petals). (*d*) The final descriptor.

is performed during the amplitude assignment into the angular and spatial bins, vii) impact of the strongest bins in weakened before the final ℓ_2 norm normalisation.

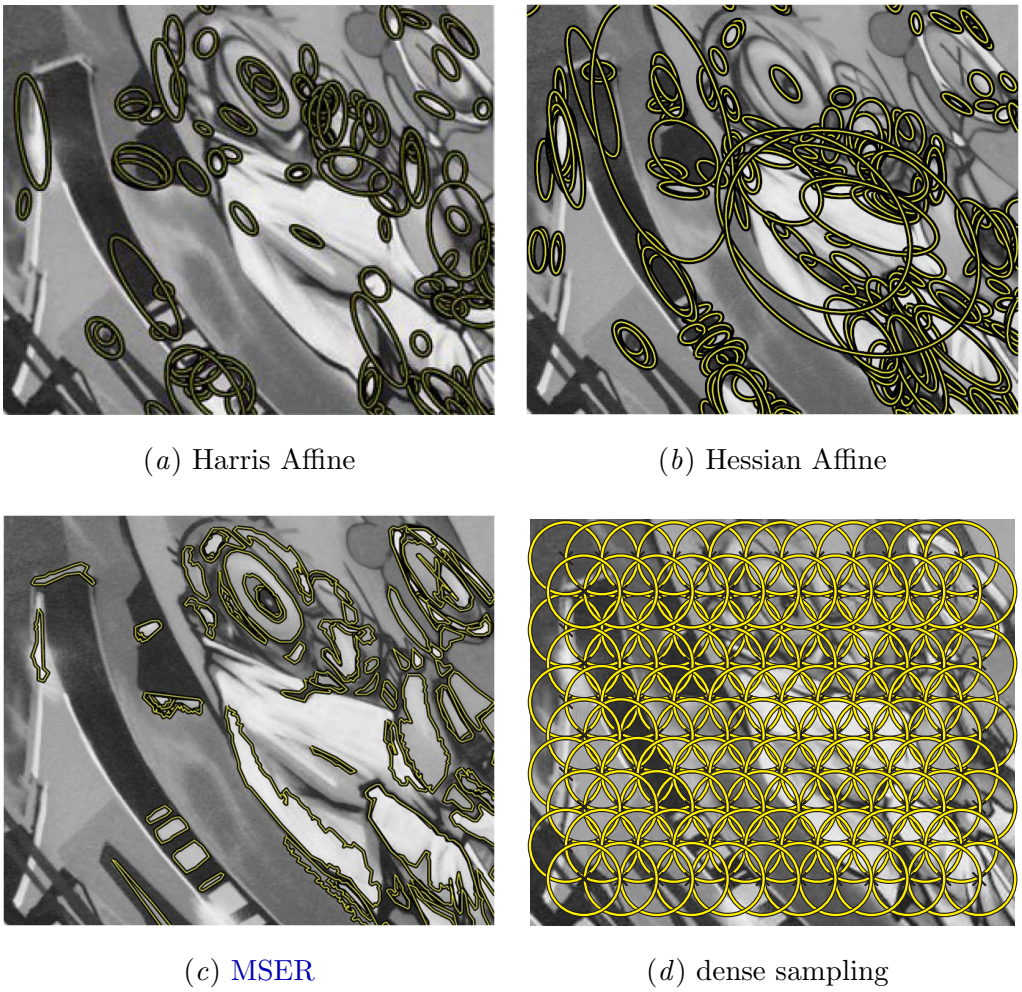
In principle, the SIFT descriptor quantifies coarsely the position, orientation, and relative strength of edges of an object captured in the patch, as illustrated in figure 1.2.

Other popular local image descriptors include Gabor Filters [Gabor, 1946, Vetterli, 1995], Histogram of Oriented Gradients [Dalal and Triggs, 2005] (HOG), Gradient Location and Orientation Histogram [Mikolajczyk and Schmid, 2005] (GLOH), Geometric Blur [Berg and Malik, 2001], Fast Local Descriptor for Dense Matching [Tola et al.,

2008] (DAISY), Speed Up Robust Features [Bay et al., 2008] (SURF), Binary Robust Independent Elementary Features [Calonder et al., 2010] (BRIEF), and many more. Several extension of SIFT descriptor that work in various colour spaces also exist. They were proposed and compared to *colour histograms* in [van de Sande et al., 2008]. Opponent SIFT is amongst the most robust colour descriptors. Also, descriptors that capture texture such as Grey-level Co-occurrence Matrices [Haralick et al., 1973] and Multi-resolution Rotation Invariant Local Binary Patterns [Ojala et al., 2002] (LBP) are widely used in VCR, especially for Face Detection and Recognition. Moreover, there exist numerous approaches to learning banks of filters on the raw pixels extracted from image patches. These filters are learnt from image patches and often comprise primitive corner-, line-, edge-, step-, and blob-like structures of various orientations. They can express contents of image patches [Roth and Black, 2005, Lee et al., 2007].

To conclude, local image descriptors and their properties of invariance are studied in depth in [Mikolajczyk and Schmid, 2005]. Moreover, a generic pipeline for customised local image descriptors is proposed in [Winder and Brown, 2007]. A number of replaceable components are suggested in their study and their best combination is determined.

Interest Point Detectors. The goal of such detectors is to provide a meaningful sampling strategy to extract image patches from an image. Most popular corner detectors include Harris Corner Detector [Harris and Stephens, 1988] and Smallest Univalue Segment Assimilating Nucleus (SUSAN) proposed in [Smith and Brady, 1997]. Other detectors tend to extract blob-like features, *e.g.* Determinant of Hessian (DoH), Difference of Gaussian (DoG), Laplactian of Gaussian (LoG) [Bretzner and Lindeberg, 1996], and based on them more recent implementations, *e.g.* SIFT detector [Lowe, 1999] (not to confuse with the SIFT descriptor), Harris-Laplace, and Hessian-Laplace detectors [Mikolajczyk et al., 2005]. These detectors can work at the selected spatial scale (uni-scale) or across multiple spatial scales (multi-scale). The latter variant is very common in VCR as the same objects often appear at various scales across collections of images. In order to make detection invariant to affine changes, Harris and Hessian Affine Region Detectors were also proposed in [Mikolajczyk et al., 2005]. Another group of region detectors is based on an unsupervised image segmentation called Watershed. Maximally Stable Extremal Regions (MSER) detector [Matas et al., 2002] selects coherent regions



(*a*) Harris Affine (*b*) Hessian Affine

(*c*) MSER (*d*) dense sampling

Figure 1.3: Examples of regions delivered by (*a*) Harris Affine, (*b*) Hessian Affine, (*c*) MSER keypoint detectors, (*d*) dense sampling (a single scale only).

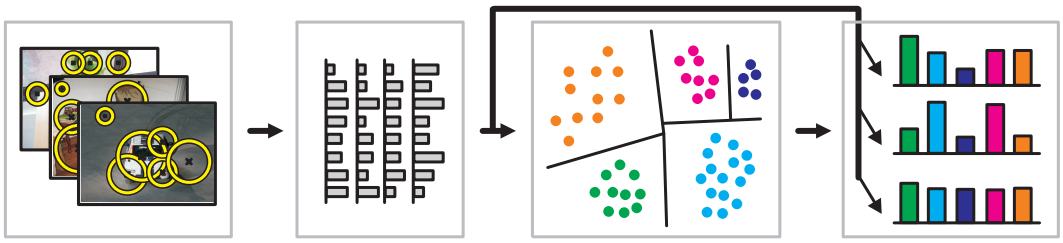
whose appearance remains sufficiently stable (not changing) over a desired number of consecutive thresholds applied to a given image (over intensity of pixels). Figure 1.3 illustrates regions obtained with Harris Affine, Hessian Affine, MSER detectors, as well as the dense sampling strategy.

To conclude, interest point detectors and their quality are studied in depth in [Mikolajczyk et al., 2005]. Colour-based interest points have been also proposed and studied in depth in [Stöttinger et al., 2012]. Moreover, several sampling strategies (keypoints as well as the dense sampling strategy) are evaluated specifically in the classification scenario [Nowak et al., 2006]. Their study advocates the dense sampling approach.

1.2.2 Image Signatures

Local image descriptors can characterise coarsely either the entire objects, the parts of objects, or the objects with fragmented surroundings depending on the scales and locations of the extracted image patches. However, simply classifying every descriptor with a classifier is an inefficient strategy as: i) such task is computationally formidable given thousands of descriptors in every image, ii) reliable object recognition often requires capturing the visual context of objects. With regards to remark (ii), if a road appears in the image context, it is likely that a car will be observed as well. If the sky appears in the image context, one can likely see a plane or a bird or the sun as well, *etc.* Similarly, various object parts may constitute the evidence of an object. For instance, if an image contains a shoe (represented by a local image descriptor), as well as a leg (another descriptor), this increases belief that the image depicts a human. Furthermore, objects of interest can appear at various positions and scales in various images. This means that often only a few of the local image descriptors from an image describe a desired object. The global scene recognition approaches are ineffective for this task as, being designed to capture only the coarse gist of an entire scene, they are not sensitive enough to the local appearances. Therefore, a trade-off between the local and global architectures seems to be the optimal strategy in the classification problems.

Bag-of-Words [Sivic and Zisserman, 2003, Csurka et al., 2004] (BoW) is a popular approach which transforms local image descriptors [Lowe, 1999, Mikolajczyk and Schmid, 2005, van de Sande et al., 2008] into image representations that are used in scene matching and classification. Its first implementations were associated with object retrieval and scene matching [Sivic and Zisserman, 2003], as well as object recognition [Csurka et al., 2004]. The BoW approach has undergone significant changes over recent years that will be addressed in further chapters of this thesis. A baseline BoW approach [Sivic and Zisserman, 2003] employs k-means clustering of local descriptors from a training dataset and assigning each descriptor to the nearest cluster. This is often referred to as Hard Quantisation or Hard Assignment. The clustering and assignment steps often vary between different models of BoW and are widely referred to as *dictionary learning* and *mid-level coding*, respectively. A histogram representing the image is obtained by



(*a*) keypoints (*b*) descriptors (*c*) k-means clustering (*d*) histograms

Figure 1.4: The basic Bag-of-Words model. (*a*) Extraction of the keypoints from the dataset (dense sampling may be applied instead). (*b*) Computation of the local descriptors. (*c*) K-means clustering of the descriptors in the high dimensional descriptor space. Often, this step employs more efficient dictionary learning (*d*) Assignment of the descriptors from individual images to the nearest clusters. This step typically results in frequency histograms of such assignments (one per image). However, alternative vectorial representations may be used.

counting the number of assignments per cluster. Averaging such counts by the number of descriptors in the image results in so-called *Average pooling* [Csurka et al., 2004, van Gemert et al., 2008, 2010]. Such an aggregation of assignments also varies between different models of BoW and is widely referred to as *the pooling step*. An image representation obtained in such a step is referred as the image signature. Investigations of the coding and pooling steps constitute an important part of this thesis. To conclude this section, figure 1.4 gives an overview of the steps involved in computation of the basic BoW approach described above.

Spatial Pyramid Matching. An additional element of the BoW approach is Spatial Pyramid Matching [Lazebnik et al., 2006] (SPM). It exploits the spatial bias in images by expressing spatial relations between the local features at multiple levels of quantisation. Once the local descriptors are extracted from an image, they are deployed across coarse-to-fine spatial windows that they fall into. Next, computations of the BoW histograms follow for every spatial window respectively. The resulting histograms are often additionally weighted. The coarser the level is the smaller the weight. When histograms from any two images are intersected to determine their similarity, such a weighting scheme results in a lesser impact of the features that are visually similar but

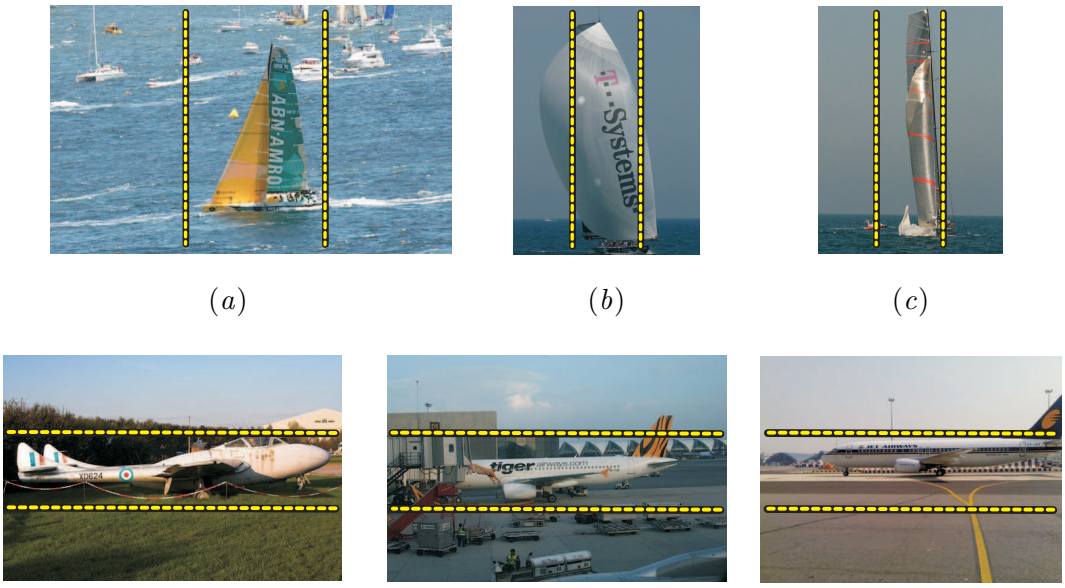


Figure 1.5: Examples of the spatial bias for (*a*-*c*) sailing boats, and (*d*-*f*) planes. Note that the middle spatial windows tend to be the most occupied with objects from the categories of interest.

spatially misaligned between these two images.

The underlying assumption of SPM is that objects of a specific class may be associated with a set of spatial positions. These objects are more likely to appear at these positions compared to other spatial locations. For instance, a plane, clouds, or the sun are likely to appear in upper parts of images while pictures of humans tend to be aligned to the middle in photographs. Such a bias is learnt from the dataset itself during the classification process. Figure 1.5 provides examples of the spatial bias for two image categories: sailing boats and planes. Note that the sailing boats and planes occupied mostly the middle vertical and horizontal spatial windows in this example, respectively.

This thesis investigates various types of bias in images that will be discussed later. Moreover, it proposes a robust alternative to the SPM scheme. By careful analysis of interaction between various stages of BoW and SPM, it is illustrated that the remarkable performance of SPM is due to additional factors beside the spatial bias.

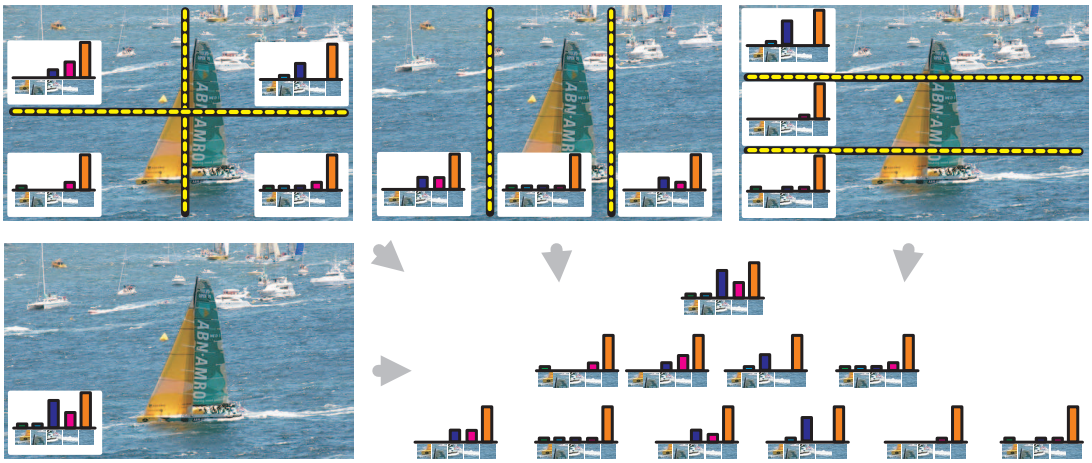


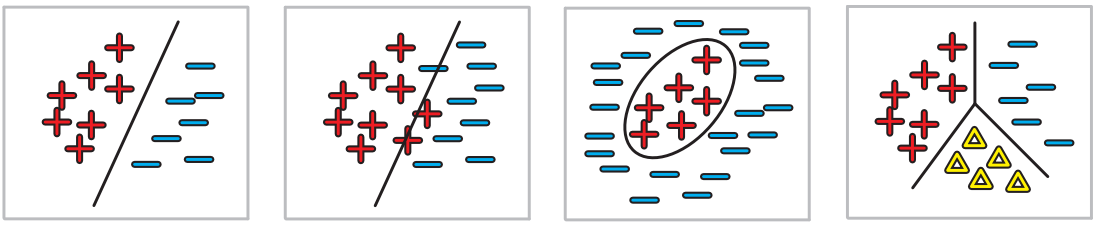
Figure 1.6: The operating principle of Spatial Pyramid Matching.

To conclude, figure 1.6 illustrates the operating principle of the SPM scheme with 1, 2×2, 3×1, and 1×3 spatial splits. Three levels of coarseness are used.

Feature Projections. For the global image descriptors, the obtained global representations often require a projection step that takes into account the class names. As global representations are highly dimensional, the projection step helps in retrieving a low dimensional manifold that represents the desired classes more accurately in a lower dimensional space. Such a manifold space is meant to provide a more meaningful similarity metric between the projected representations. To conclude, various dimensionality reduction approaches are compared in [Song and Dacheng, 2010]. Such projections are also plausible for the BoW model. However, the manifold learning is often performed in the coding step of BoW making projections somewhat redundant.

1.2.3 Image Classification

The role of a classifier is to learn how to separate several classes of interest in the feature space containing the image signatures (multidimensional vectors), and to reliably predict the class labels for previously unseen images. The quality of a classifier depends on how well it generalises from training to correctly classify the unseen instances. A classifier performs the classification task on the image signatures or so-called *kernel matrix* (or *kernel* for short). Depending on the classifier type (*linear* or *non-linear*), a decision boundary separating two classes of features may be of linear or non-linear



(a) (b) (c) (d)

Figure 1.7: Various classification problems. A linearly (*a*) separable and (*b*) inseparable two class problem. (*c*) Non-linear classifier. (*d*) Multi-class classifier.

nature. This is illustrated in figures 1.7 (a, c). The linear decision boundary is often unable to fully separate numerous samples from different classes as shown in figure 1.7 (b). The linear classifiers that employ kernels allow both linear and non-linear classification if a non-linear kernel is employed. The latter case often leads to better classification results. The linear classifiers that use the image signatures perform the linear classification only. They may be less accurate but are very efficient computationally. Many classifiers are typically *the binary classifiers* as they distinguish between two classes of interest (so-called *two-class problem*). This is shown in plots 1.7 (a-c).

Multi-class vs Multi-label. There exist fundamental differences between the data labelling processes for the VCR datasets. Some sets contain only one kind of object of interest per image. These sets are classified with so-called *multi-class classifiers*. Datasets that contain many kinds of objects of interest per image are classified with *multi-label classifiers*. The multi-class classification can be performed by several binary classifiers. Each classifier is trained for one class against the rest (so-called *one-vs-all* strategy). Then, the strongest responding classifier determines the class of an image. Also, there exist explicitly designed multi-class classifiers (as opposed to the fusion of binary classifiers) which take advantage of a constraint that only one class of objects can appear in an image. Figure 1.7 (d) illustrates this. The multi-label classifiers also often employ the one-vs-all strategy. Every class is trained against the rest, then, each classifier specifies if a class that it was trained for has been observed in an image.

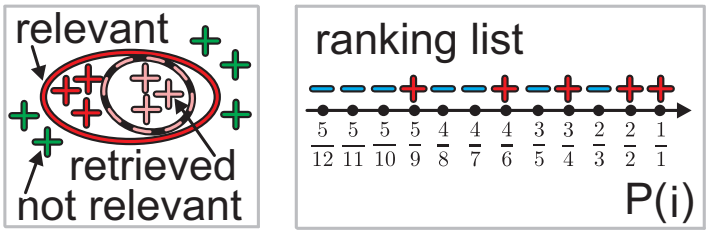
Training, Validation, and Testing. The practical classification step consists of *training* and *validation* of the model on the training and validation sets of image signatures selected for this procedure. This way the classifier is trained and its parameters are fine-tuned for the best performance. Consecutively, these training and validation sets are merged together and training is performed on the resulting set given the parameters estimated during the validation step. Finally, *testing* on a previously unseen testing set is performed in accordance with the best practice [Everingham et al., 2007].

Popular Classifiers. There exist many kinds of classifiers that can be used for this final step, *e.g.* Support Vector Machine (SVM) proposed by [Cortes and Vapnik, 1995], Linear Discriminant Analysis (LDA) proposed in [Fisher, 1936] and outlined in [Duda et al., 2001], their kernelised versions such as SVM (dual form) and Kernel Fisher Discriminant Analysis [Mika et al., 1999] (KDA) allowing non-linear classification due to the kernel trick [Aizerman et al., 1964], the multi-class equivalent of LDA first proposed in [Rao, 1948] and extended to the multi-class KDA based on Spectral Regression in [Cai et al., 2007]. The family of classifiers also entails Naive Bayes Classifier [Domingos and Pazzani, 1997], Quadratic Classifiers, Boosting [Schapire, 1990], Decision Trees [Quinlan, 1986], Random Forests [Breiman, 2001], and many others.

In this thesis, SVM [Chang and Lin, 2011] and multi-class and multi-label KDA [Tahir et al., 2009, 2010] classifier implementations are used. These classifiers are often combined with either the linear or Radial Basis Function (RBF) kernels [Scholkopf et al., 1997]. The exact classification arrangements are explained in every chapter for clarity.

1.2.4 Performance Measures

A binary classifier can output two types of predictions: i) binary class predictions that indicate for every image if it contains any instances of the positive category, ii) probabilistic scores (or ranking list) that reflect for every image the likelihood that it includes at least one instance from the positive category.



(a) (b)

Figure 1.8: Performance measures. (*a*) Accuracy is defined as the ratio of the retrieved relevant positive to the relevant positive instances. (*b*) Average Precision requires the ranking list with Precision denoted as $P(i)$ and computed at cut-off $i = 1, 2, ...$

The classification performance can be characterised by two measures:

$$\text{Precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|} \tag{1.1}$$

$$\text{Recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|} \tag{1.2}$$

For the visual categorisation problems, the relevant documents of class $c \in \mathcal{C}$ are defined as images each containing at least one instance of class c. However, definitions of the retrieved documents vary. Given cases (i) and (ii), the retrieved documents are: i) images considered by the classifier to contain at least one instance of class $c \in \mathcal{C}$, ii) all images processed by the classifier.

Multi-class Problems. For the multi-class predictions, the classifier makes binary decisions about the classes as defined in case (i). The Recall scores are computed accordingly for every class $c \in \mathcal{C}$. They are referred to as the Accuracy scores. One can also define Accuracy corresponding to Recall as the number of images that are correctly predicted by the classifier to contain instances of class c, divided by the number of images each truly containing at least one instance of class c. This is illustrated in figure 1.8 (a). The Accuracy score for the example in the plot is $\frac{3}{6}$. Moreover, if the classifier was to label all images as containing instances of class c, the Accuracy score for class c would amount to 1. This would be a bad indicator of the classification quality. However, the Accuracy scores for classes $c' \neq c$ would amount to 0 in such a case. Thus, *Mean Accuracy* is a single relevance score defined as the average of all Accuracy scores. For simplicity, Mean Accuracy is referred to as *accuracy* in the following chapters.

Multi-label Problems. *Average Precision* is a popular measure for multi-label problems. It takes into account both Recall and Precision. As the classifier is not required to make any hard decisions for this measure, the order of the positive instances in the ranking list determines how well the positive and negative instances in one-vs-all problem can be linearly separated from each other in this list. If full separability is achievable, the score is 1. Formally, Average Precision is defined as the area under Precision vs Recall curve $p(r)$ for case (ii), and computed for a given one-vs-all problem:

$$\mathrm{AP} = \int_0^1 p(r)dr \tag{1.3}$$

As datasets provide discrete instances of class $c \in \mathcal{C}$ (vs other instances), the integration in formula (1.3) is replaced with a finite sum over ranked images:

$$\mathrm{AP} = \sum_{i=1}^{|\mathcal{I}_c|} P(i)\Delta r(i) = \frac{\sum_{i=1}^{|\mathcal{I}_c|} P(i) \cdot Pos(i)}{|\{\text{relevant documents}\}|} \tag{1.4}$$

Variable i is the rank in sequence $\mathcal{I}_c$ of retrieved images while $|\mathcal{I}_c|$ is the number of all retrieved images such that $|\mathcal{I}_c| = |\mathcal{I}|$, where $\mathcal{I}$ is the entire image set. Symbol $P(i)$ denotes Precision computed at a cut-off i in sequence $\mathcal{I}_c$. Symbol $\Delta r(i)$ is a change in Recall from step $i-1$ to i that can be also defined as $\Delta r(i) = \frac{Pos(i)}{|\{\text{relevant documents}\}|}$. $Pos(i) = 1$ if the retrieved image at position i in sequence $\mathcal{I}_c$ contains at least one instance of class $c \in \mathcal{C}$ (a true positive), $Pos(i)=0$ otherwise. Figure 1.8 (b) illustrates an arbitrary ranking list with the corresponding values of $P(i)$. Average Precision for this example amounts to $\left(\frac{1}{1}+\frac{2}{2}+\frac{3}{4}+\frac{4}{6}+\frac{5}{9}\right)/5 \approx 0.794$.

Furthermore, *Mean Average Precision* (MAP) is a single relevance score defined as the average over all Average Precision scores (one per class). This measure is used in this thesis for the multi-label problems. For reference, the above measures are comprehensively described in [Zhu, 2004].

1.3 Challenges

Visual categorisation faces a number of challenges due to the high dimensional nature of images, small amounts of training samples (labelling is time-consuming), and varying

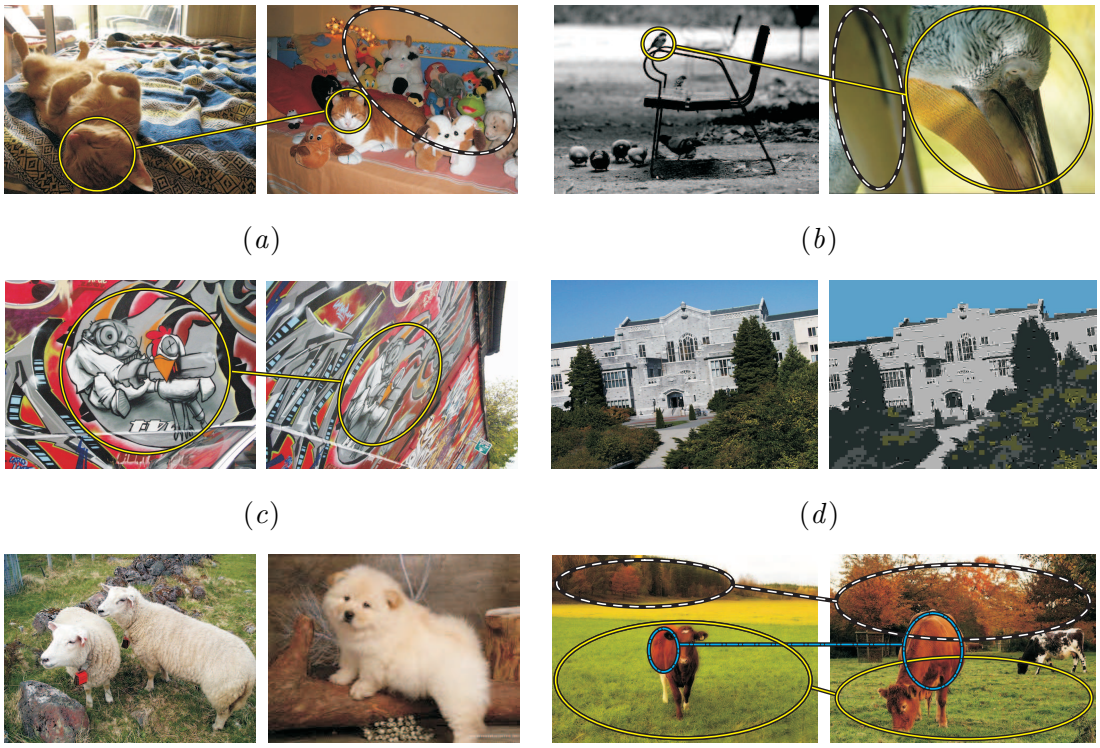


Figure 1.9: Illustration of challenges in VCR. (a) Rotation and scale changes are circled with solid lines. Background clutter is represented by a dashed circle. (b) Extreme scale change and undesired intra-class variability are circled with solid lines. Partial occlusion is shown with a dashed ellipse. (c) Viewpoint changes. (d) Compression artifacts. (e) Obstructive inter-class similarity between sheep and a Chow Chow dog. (f) Areas spanned by repetitive visual patterns of the same types vary in their sizes.

difficulty of visual concepts. Below, the notorious challenges are briefed in a list while their examples are illustrated in figure 1.9:

- *Geometric and Photometric Image Transformations.* Objects undergo various transformations such as rotation, scale and viewpoint changes, translation, brightness, blur, and colour changes. Acquisition noise, lens distortions, and compression artifacts also pose problems. These transformations result in large variations between the signals expressed by the image representations. Thus, mathematical algorithms face difficulty in generalising given such signals, *e.g.* a vehicle occupying an entire image differs from a car in a background (a few pixels wide).

- *Background Clutter, Occlusions, and Truncation.* Objects often appear in various backgrounds with substantial clutter. These undesired stimuli can trick detectors into detecting objects that are not in the image due to similarities of the appearances. Moreover, clutter can occlude objects. While humans cope with occlusions by understanding the context and anatomy of objects, mathematical models are still short of efficient mechanisms coping with such a phenomenon. Similarly, digital images often contain objects that are truncated. This poses similar challenges to occlusions.

- *Varying Context.* Image background often contributes to the scene understanding. For instance, roads and buildings increase likelihood of presence of cars. A similar is true for the foregrounds, *e.g.* a wheel may suggest appearance of other visible parts of a vehicle. However, with strong variations in the context that can happen naturally, recognition algorithms often fail.

- *Depictive Styles.* Images may differ in their depictive styles which determine whether an object is photographed or painted or drawn. Depending on style, signals captured in the image representations may differ significantly.

- *Intra-class Variability.* Intra-class variability is concerned with large variations between objects (or visual concepts) of the same class. For instance, Chow Chow, Bull Terrier, and Airedale Terrier dogs are visually quite different. Also, a jumbo jet and the Su-47 fighter planes are unlike each other. Despite their differences, the image representations have to be invariant enough to the variations and the classifier has to generalise well to mitigate the differences at the recognition stage.

- *Inter-class Variability.* Inter-class variability is concerned with small variations between objects (or visual concepts) from different classes. Objects (or visual concepts) from several different classes may be more visually similar to each other than to objects (or visual concepts) representing the same category. For instance, a Chow Chow dog may be easily confused with a sheep due to their white woolly appearances and similar body postures. In fact, cats and dogs represent two categories that are often confused with each other as visual differences between these two species are very subtle from the algorithmic point of view.

- *Computational Complexity.* Algorithms have to cope with the above challenges to provide a reliable image categorisation. The price for dealing with complex visual scenes and difficult object taxonomies is a large computational complexity.

Although the above challenges have been addressed to a certain degree in the variety of studies referenced in this thesis, they nonetheless are active topics of research in VCR. Moreover, there exist yet another known challenge in visual categorisation, only recently brought to attention in BoW, that is of paramount interest to this thesis:

- *Repetitive Visual Stimuli.* Repetitive visual patterns of any given appearance are present in varying quantities across images. For instance, areas spanned by the natural landscapes vary. Grasses, rocks, sand, reservoirs of water, the sky, foliage, and other vegetation, all can appear in unpredictable quantities in images. This is illustrated in figure 1.9 (f). Similar holds true for the urban scenery. Brick walls, windows, tarmac, cobbles, and pavements can span across unpredictable areas. Moreover, this also holds true for images that are taken in other uncontrolled environments. As it is explained below, this unpredictability introduces a harmful variance into the image signatures produced by the baseline BoW model from section 1.2.2.

Baseline BoW assumes that each visual word in a visual dictionary is associated with a visual appearance of some kind. Moreover, this model counts occurrences of visual words for any given type that are voted for by the local descriptors extracted from an image. If such descriptors are extracted numerous times from a repetitive visual pattern like a field of grass due to the dense sampling strategy or an interest point detector firing multiple keypoints, a visual word representing patch of grass will be voted for multiple times. Therefore, such a phenomenon introduces large variances in the counts of visual words. This thesis proposes a number of novel image representations with the goal of limiting this undesired phenomenon, as explained next in the list of contributions.

1.4 Publications

This thesis builds upon publications prepared in the course of my PhD studies:

- P. Koniusz and K. Mikolajczyk. Segmentation Based Interest Points and Evaluation of Unsupervised Image Segmentation Methods. *BMVC*, 2009
- P. Koniusz and K. Mikolajczyk. On a Quest for Image Descriptors Based on Unsupervised Segmentation Maps. *ICPR*, 0:762–765, 2010. ISSN 1051-4651
- P. Koniusz and K. Mikolajczyk. Soft Assignment of Visual Words as Linear Coordinate Coding and Optimisation of its Reconstruction Error. *ICIP*, 2011a
- P. Koniusz and K. Mikolajczyk. Spatial Coordinate Coding to Reduce Histogram Representations, Dominant Angle and Colour Pyramid Match. *ICIP*, 2011b
- P. Koniusz, F. Yan, and K. Mikolajczyk. Comparison of Mid-Level Feature Coding Approaches And Pooling Strategies in Visual Concept Detection. *CVIU*, 2012. ISSN 1077-3142. doi: 10.1016/j.cviu.2012.10.010
- P. Koniusz, F. Yan, P. Gosselin, and K. Mikolajczyk. Higher-order Occurrence Pooling on Mid- and Low-level Features: Visual Concept Detection. *PAMI*, 2013. (submitted)
- M. Barnard, P. Koniusz, W. Wang, J. Kittler, S. M. Naqvi, and J. Chambers. A Robust Approach to Joint Audio-Visual Tracking Based on Bags of Visual Words. *TMM*, 2013. (submitted)
- M. A. Tahir, F. Yan, P. Koniusz, M. Awais, M. Barnard, K. Mikolajczyk, and J. Kittler. A Robust and Scalable Visual Category and Action Recognition System using Kernel Discriminant Analysis with Spectral Regression. *TMM*, 2012

Other achievements relevant to this thesis include:

- First prize for SURREY_MK_KDA system that scored the highest MAP of 62.15% amongst competing approaches in the PASCAL VOC2010 Action Classification Teaser Challenge [Everingham et al., 2010].
- An Outstanding Reviewer Award for BMVC 2012 [Bowden et al., 2012].

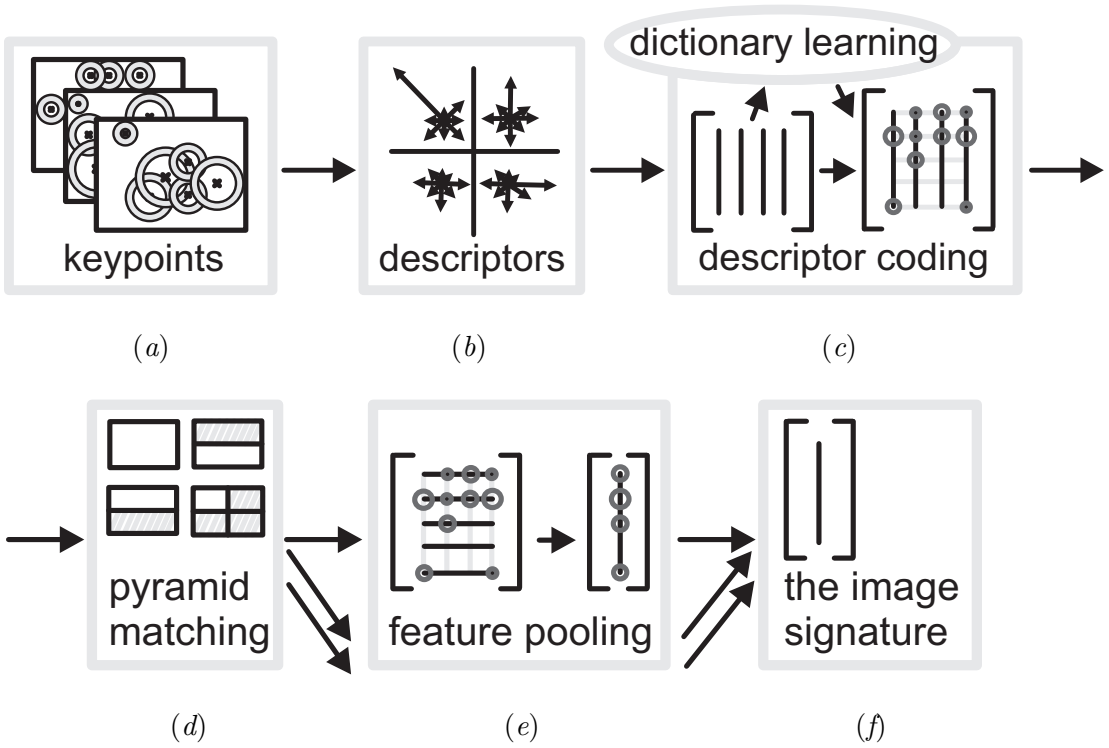


Figure 1.10: Various steps constituting on Bag-of-Words. We investigate (*a*) keypoint design, (*b*) descriptor design, (*c*) various coding techniques, (*d*) pyramid matching schemes, (*e*) mid-level feature pooling. (*f*) The image signatures are fed to a classifier.

1.5 Contributions and Thesis Structure

In this thesis, a number of contributions are made with respect to the design of novel and robust image representations for visual categorisation. The building blocks of the BoW model in figure 1.10 are investigated. This model represents a more detailed diagram of basic BoW from figure 1.4. The chapters follow the illustrated steps from left to right and address the following aspects: the keypoint design (a), the descriptor design (b), various coding techniques (c), alternative pyramid matching schemes (d), improved pooling approaches (e) that result in the image signatures (f). The detailed role of the above steps beyond the introduction from section 1.2 will be explained in the corresponding chapters. The proposed approaches lead to improvements over the state-of-the-art systems which are reported in evaluations at the end of every chapter. This thesis is structured around the following list of contributions:

I. An interest point detector based on the unsupervised image segmentation maps is proposed. This is a corner detector that operates on the junctions along boundaries between segments. Therefore, the keypoints from uniform uninformative parts of visual scenes are suppressed and the main attention is given to the visually relevant regions. Moreover, the corners of segments are evaluated and found to be repeatable features in segmentation maps. Therefore, an evaluation of the unsupervised image segmentations is proposed based on the corner features. Their utility in visual categorisation is also evaluated. The results on BoW with the local image descriptors extracted in this manner show a promising improvement. This work was published in [Koniusz and Mikolajczyk, 2009].

   Chapter 2 outlines the proposed interest point detector and provides necessary evaluations for the employed segmentation algorithms.

II. Segmentation-based image descriptors are proposed for Visual Object Category Recognition. In contrast to commonly used interest points, the proposed descriptors are extracted from pairs of adjacent regions given by an unsupervised segmentation method. In this way, semi-local structural information from images is exploited. The segments are used as spatial bins of descriptors. This eliminates multiple contributions form large uniform regions. Image statistics based on gradient, colour, and region shape are extracted over corresponding regions in images. The proposed descriptors are evaluated on standard recognition benchmarks. Results show they outperform state-of-the-art reference descriptors with 5.6× less data. This work was published in [Koniusz and Mikolajczyk, 2010].

   Chapter 3 introduces the segmentation-based image descriptors as well as the performed experiments.

III. A highly popular technique for coding the local image descriptors in the BoW model, called *Visual Word Uncertainty* (VWU) or *Soft Assignment* (SA) that was proposed in [van Gemert et al., 2010], is combined with *Linear Coordinate Coding* (LCC) studied in [Yu et al., 2009]. As a contribution, it is shown that SA, an approach derived from Gaussian Mixture Model (GMM), can act as an approximation to the LCC methods by combining SA with the *quantisation loss*

used by the LCC coders. An optimisation is performed over the smoothing factor of the SA model. Minimising the quantisation loss is demonstrated to correlate well with the best classification performance. This work was published in [Koniusz and Mikolajczyk, 2011a].

Chapter 4 demonstrates that the SA coding approach can act as an approximation to the LCC methods.

IV. An alternative approach to SPM that introduces spatial information to the BoW model, called *Spatial Coordinate Coding* (SCC), is proposed. It reduces the sizes of image signatures tenfold compared to SPM and decreases computational and memory requirements. Specifically, spatial locations of image patches are added at the descriptor level. Hybrids between the proposed model and SPM are also studied. Moreover, Pyramid Matching is successfully applied to measurements such as dominant orientations of edges and colour, resulting in *Dominant Angle Pyramid Matching* (DoPM) and *Colour Pyramid Matching* (CoPM) approaches. This work was published in [Koniusz and Mikolajczyk, 2011b].

Chapter 5 introduces the SCC, DoPM, and CoPM approaches.

V. In the BoW model, the local descriptors are extracted from images and expressed as vectors representing visual word occurrences, referred to as *mid-level features*. Various methods for generating mid-level features, including Soft Assignment, *Locality-constrained Linear Coding* (LLC), and *Sparse Coding* (SC) are reviewed. A fast coder called *Approximate Locality-constrained Soft Assignment* (LcSA) is proposed, its quantisation loss is optimised, and its relation to LLC is shown.

Moreover, various *pooling methods* that aggregate mid-level features into vectors representing images are investigated, including Average pooling, Max-pooling, and a family of likelihood inspired operators. Interactions between both coding schemes and pooling methods are demonstrated.

Furthermore, a generalisation of the investigated pooling methods that accounts for *the descriptor interdependence* is proposed and an improved pooling that addresses noise effects in mid-level features is introduced. An efficient approach for coding is developed. This work was published in [Koniusz et al., 2012].

Chapter 6 introduces SA, SC, and LLC coding, as well as the proposed LcSA coder. The pooling operators and the proposed pooling improvements are introduced. Finally, extensive evaluations for the mid-level coding and pooling approaches are provided.

VI. In the BoW model, the statistics are extracted from mid-level features with a pooling operator. As pooling typically aggregates only occurrences of visual words represented by coefficients of each mid-level feature vector, it produces the first-order statistics only. Therefore, to employ the more informative second- or higher-order statistics, aggregation over co-occurrences or higher-order occurrences of visual words in mid-level features. Moreover, a relevant derivation based on kernel linearisation is proposed and a generalisation to various pooling operators is exploited: Average, Max-pooling, Analytical pooling, and a highly effective trade-off between Max-pooling and Analytical pooling. For bi- and multi-modal coding with two or more coders, an extension also based on kernel linearisation is derived.

Moreover, it is demonstrated by combining both the grey scale and colour mid-level features that such a linearisation outperforms naive fusing schemes. An explanation is given that the SPM scheme in BoW and other similar methods are robust performers as, being special cases of the proposed method, they produce the second- rather than first-order statistics. Moreover, a Residual Descriptor that exploits the quantisation loss in coding is designed for the bi-modal extension. Comparisons to state-of-the-art methods are provided. This work is presented in [Koniusz et al., 2013].

Chapter 7 explains the proposed aggregation step over co-occurrences of visual words in mid-level features called Second-order Occurrence Pooling. Higher order statistics are also evaluated. An extension to bi- and multi-modal coding is evaluated on the grey scale and colour features, as well as the Residual Descriptor.

Finally, chapter 8 concludes this work and reflects on ideas for the further research.

Chapter 2

Segmentation Based Interest Points

This chapter investigates segmentation based interest points for matching and recognition. We propose two simple methods for extracting features from the segmentation maps, which focus on the boundaries and centres of the gravity of the segments. Moreover, our evaluations provide a new insight into suitability of the segmentation methods for generating local features for image retrieval and recognition. Several segmentation methods are evaluated and compared to state-of-the art interest point detectors using the repeatability criteria as well as matching and recognition. In addition, we propose to measure the robustness of segmentations by the repeatability of features extracted from segments on images distorted by various geometric and photometric transformations. Typical evaluations quantify separability of foregrounds from backgrounds.

2.1 Introduction

One of the crucial issues in image retrieval or recognition is the extraction of salient features. Segmentation methods seem to have great potential of delivering good features as their main goal is to separate foreground objects from backgrounds. For instance, in [Russel et al., 2006], multiple segmentations were used to find objects and their extent in collections of images. The underlying assumption was that all similar objects across

images give rise to segments alike, and those irrelevant appear dissimilar. Reminiscent approaches were taken in [Malisiewicz and Efros, 2007]. Their work concluded that even over-complete representations may be insufficient to achieve satisfactory repeatability of segmentation maps. Similar scenes affected by natural lighting conditions, angle of view and scale result in different segmentation maps. Thus, partial matching was taken into further investigation in [Hedau et al., 2008]. We argue that stability of produced partitions is more important than unambiguous foreground vs background separation for such applications. The approaches taken in [Russel et al., 2006, Malisiewicz and Efros, 2007, Hedau et al., 2008] show that segmentation methods can be used in VCR.

This chapter reports on a set of tests which aimed at identifying what kinds of features from general-purpose segmentation algorithms are stable. This enables further exploitation of these stable parts to build reliable representations for image content retrieval or classification systems. We are unaware of any previous evaluation that targets stability of segmentations with the use of interest points and recognition, which makes this work novel in these areas and contributes to the segmentation evaluation problem. Furthermore, it also contributes towards bridging the gap between the interest point detectors and unsupervised segmentations. Along with a simple testing protocol, we propose interest point detectors based on the segmentation maps that may be directly used in many applications utilising interest points. An extensive evaluation demonstrates the performance of these interest point detectors. This characterises the quality of different segmentations. In contrast to the existing evaluations like [Ge et al., 2006, Martin et al., 2001, Arbelaez et al., 2007], we quantify the performance of segmentation methods in terms of suitability for recognition with means of the local descriptors. Now, a brief review of evaluation benchmarks for the interest point detectors and unsupervised image segmentations will be given, respectively.

2.1.1 Benchmarks for Interest Point Detectors

An introduction to the state-of-the-art interest point detectors is provided in section 1.2.1. A variety of such methods are based on the corner, blob, region, or saliency driven detection on the contour-, intensity-, or parametrisation-based representations of

images. Single- and multi-scale, as well as affine extensions are popular. An exhaustive survey of state-of-the-art keypoint detectors can be found in [Mikolajczyk et al., 2005, Tuytelaars and Mikolajczyk, 2008]. The interest points are typically characterised in terms of repeatability and invariance to different geometric and photometric changes. A very popular testing approach is based on *the repeatability* of detected features between the reference and transformed images that are related by a homography matrix. The repeatability is a relative measure counting the matched features to the total number of detected features. The keypoints that are matched given an image transformation are known as *the correspondences*. One can argue whether detectors that produce highly repetitive keypoints and small counts of correspondences have any practical use in visual categorisation. In contrast, the dense sampling strategy offers an excellent coverage of scenes in images. However, dense sampling combined with the local image descriptors results in a high number of mismatches between images for the Nearest Neighbour (NN) matching strategy. Hessian and Harris affine detectors, as well as MSER [Matas et al., 2002], are the best performing detectors according to [Mikolajczyk et al., 2005]. Other benchmarks of the interest point detectors employ:

- The ground-truth verification that quantifies missed features and false positives.
- The visual inspection [Lopez et al., 1999] with a set of visual quality criteria.
- The localisation accuracy [Heyden and Rohr, 1996] to determine how accurate are coordinates of keypoints given two images related by a homography. Such a criterion is somewhat complementary to *the overlap* measure [Mikolajczyk et al., 2005] that counts matched features that overlap with each other at least partially.
- The information content criterion introduced in [Schmid et al., 2000]. It quantifies how distinctive are the local image descriptors extracted at any given keypoint location, compared to the rest of such extracted descriptors.

2.1.2 Benchmarks for Unsupervised Segmentations

According to a recent survey on quality of unsupervised segmentations [Ge et al., 2006], the most robust approaches are *Mean Shift* [Comaniciu and Meer, 2002, 2003] (MS), *Ef-*

ficient Graph-Based Image Segmentation [Felzenszwalb and Huttenlocher, 2005] (EGO), and Normalised Cuts [Shi and Malik, 2002, Cour et al., 2004] (NC). The results in their benchmark were obtained on a dataset of 1023 images by evaluating so-called Upper-Bound Performance [Ge et al., 2006] gauging how well resulting segments adhered to the contours separating unambiguously defined ground truth foregrounds from backgrounds. This criterion appeared to be biased towards overly small structures but worked well for extremely large segments. Combined performance of segmentations was estimated and their complementary nature emphasised. The dataset from their studies contains the grey scale images at two resolutions: 80×80 and 200×200 pixels.

Another benchmark proposed in [Martin et al., 2001] evaluates how well segments adhere to the regions from multiple maps annotated by various human subjects. Such maps may exhibit different levels of refinement, *e.g.* an outline of a head, eyes, mouth, hairline represent a refinement of a head. This is a paradigm shift in the scoring criterion from a single to multiple acceptable segmentations per object, respectively. Their dataset consists of $12K$ manually annotated segmentations of $1K$ images from Corel dataset [Arbelaez et al., 2007]. Their evaluation builds on Local Refinement Error which estimates how many pixels belong to a given ground truth region R in segmentation map S_1 and do not belong to the corresponding region R in segmentation map S_2, all normalised by the total number of pixels in region R of map S_1. Moreover, Global Consistency Error expects that refinements of regions R_n can take place either in map S_1 or S_2 (not in both). Local Consistency Error allows mixed refinements, some taking place in map S_1, other in S_2. These measures result in low errors if one of the two compared segmentations is just a refinement or generalisation of the other map.

Precision-recall curves given a measure of matched pixels from boundaries between two segmentations were applied in [Estrada and Jepson, 2005]. The best performers were: SE Min-Cut, Canny Edge Detector, Mean Shift, Local Variation, and Normalised Cuts.

Lastly, recent survey proposed in [Zhang et al., 2008] reviewed various benchmarks for evaluating segmentations and compared their pros and cons. It was pointed out that these benchmarks are reliable for specific segmentation methods they were designed to work with. The objectivity of such benchmarks was questionable otherwise.

2.2 Proposed Interest Point Detectors

This section briefly discusses the investigated segmentation approaches and then presents the methods for extracting local features from their segmentation maps.

2.2.1 Unsupervised Segmentation Methods

This study follows the findings of [Ge et al., 2006] and focuses on measuring performance of Efficient Graph-Based Image Segmentation (EGO), Mean Shift (MS), Watershed (WA), and Normalised Cuts (NC) in terms of their stability. Figure 2.1 provides illustrations for the described below segmentation algorithms.

EGO [Felzenszwalb and Huttenlocher, 2005] is a graph-based technique where all vertices represent pixel coordinates of an image and edges represent a similarity measure between neighbouring vertices by the difference of the colour channels. This method

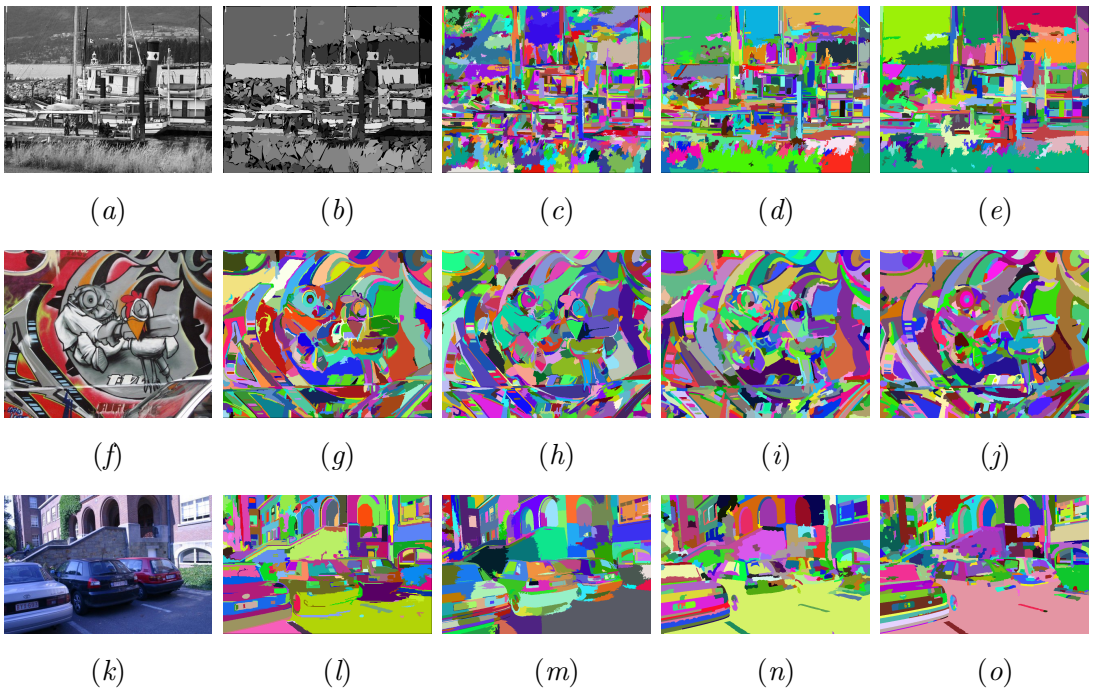


Figure 2.1: Example images (*a*, *f*, *k*) with the corresponding segmentation maps. (*b*) An EGO segmentation map approximated with polygons. (*Top*) Results of EGO for over-, well-, and under-segmented sets (*c*-*e*). (*Middle*) Segmentation maps from the EGO, MS, NC, and WA methods (*g*-*j*). (*Bottom*) More maps for these methods (*l*-*o*).

employs the bottom-up merging strategy based on a pair-wise region comparison.

MS [Comaniciu and Meer, 2002] is based on a connectedness criterion. Image pixels are considered as vectors in $5D$ space of spatial and colour coordinates. A centroid-based mode detection is employed and the coordinates are ascribed to the modes. A recursive fusion of the basins of attraction merges the modes located within a certain radius.

NC [Shi and Malik, 2002] is also based on a graph of vertices and edges representing pixel coordinates and their similarities. Such a a graph stores $N \times (width \times height)^2$ bytes of data. N represents the size of the weight coefficients (given in bytes). Image partitioning is performed by a cut between two disjoint sets of vertices which optimise the normalised cost criterion. We modified the segmentation process to overcome the complexity issues due to which images larger than 200×200 pixels cannot be easily handled. Moreover, the performance of the original implementation tended to degrade in presence of scale and affine changes. Therefore, larger images were split into a set of half-overlapping sub-windows. The resulting segments from over-segmented maps were merged by using only those non adjacent to the boundaries of sub-windows to avoid artifacts. Note that the segments adjacent to the boundary of a sub-window have their undistorted segment counterparts in another shifted sub-window. A merging strategy similar to the WA post-processing described further in the text was applied. This gave satisfactory segmentation results and significantly reduced the processing time.

WA [Ibanez et al., 2005] segmentation acts on the image luminance or colour maps and uses the gradient descent to seek for local minima. Thus, the pixels are attracted to the minima within a given basin of attraction. This method benefits from combining it with an anisotropic filtering introduced in [Perona and Malik, 1990]. We introduced an additional post-processing step by sorting all segments in the ascending order by size and merging first K percent of adjacent small segments based on their similarity.

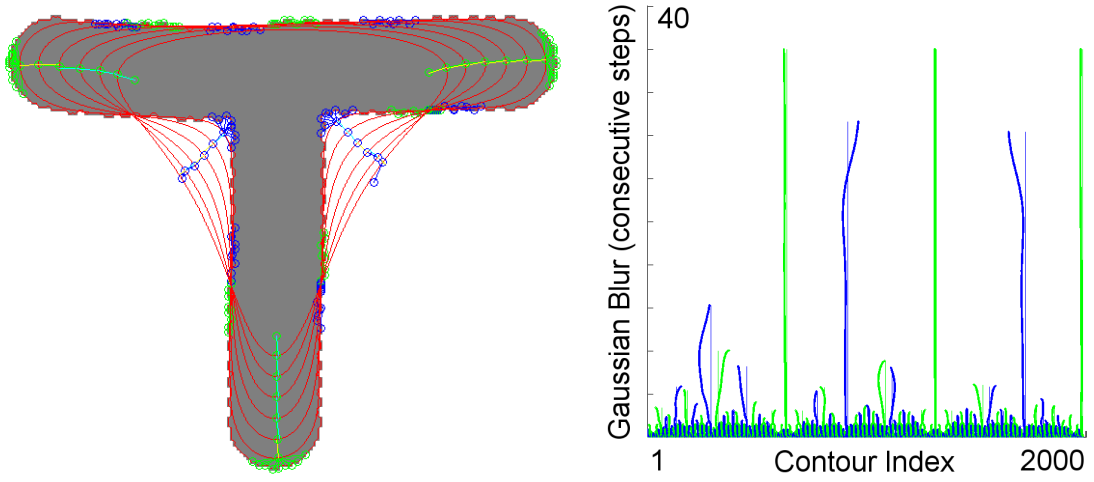
2.2.2 Detection of Interest Points from Segmentation Maps

Inspired by the evaluation of the affine region detectors [Mikolajczyk et al., 2005], this study focused on two kinds of keypoints locating potentially salient parts of segments. Moreover, three different interest point detectors were devised.

Ellipses inscribed in the segments are potentially repeatable features. Estimation of centres and fitting the ellipses can be performed on either contour coordinates or over the whole area. We found that ellipses fitted to the areas of segments are more repeatable than contour-based variants, as the segments often suffer from partial spilling into thin branch-like noisy structures under geometric and photometric changes.

Corners located on the boundaries between regions are salient features. They help overcome the structural noise of segmentations as partial spilling of segments affects only a fraction of such corners. The scale-space theory and the curvature measure researched in [Mokhatarian et al., 1996] were applied to contours extracted from segments. Therefore, the maximally concave and convex points were extracted from the contours. This method is illustrated in figure 2.2. In more detail, the contour-based interest point detector extracts the spatial coordinates from segments and normalises them to contain $N = 2000$ samples. This results in a vector of coordinates per segment:

$$[\boldsymbol{x}\ \boldsymbol{y}] = \begin{bmatrix} x_1,\ \ldots,\ x_N \\ y_1,\ \ldots,\ y_N \end{bmatrix}^T \tag{2.1}$$



(*a*) (*b*)

Figure 2.2: (*a*) A T-shaped segment, its contour, and consecutive contours resulting from coordinate blurring, and the extreme curvature points tracked over the scale-space. The maximally concave and convex points are in blue and green. (*b*) The maximally concave and convex points as a function of the contour index and the blurring step.

The scale-space scheme is applied to vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. For this purpose, two convolutions with a Gaussian mask are applied to $\boldsymbol{x}$ and $\boldsymbol{y}$ given a desired standard deviation σ. These operations are performed with the modulo arithmetic. Such a blurring results in smoothing and shrinkage of contours. For these experiments, standard deviations were set to $\sigma = 5$, 10, ..., 200, resulting in 40 blurring steps. The curvature measure is then applied to each of smoothed contours represented by vectors $\boldsymbol{x}_\sigma$ and $\boldsymbol{y}_\sigma$:

$$\boldsymbol{k}_\sigma(\boldsymbol{x}_\sigma, \boldsymbol{y}_\sigma) = \frac{\boldsymbol{x}'_\sigma \boldsymbol{y}''_\sigma - \boldsymbol{y}'_\sigma \boldsymbol{x}''_\sigma}{[(\boldsymbol{x}'_\sigma)^2 + (\boldsymbol{y}'_\sigma)^2]^{3/2}} \tag{2.2}$$

The first and second derivatives are denoted as $'$ and $''$. Note that the operations of raising to the power of 2 and 3/2 are element-wise. Values of such a curvature measure such that $\boldsymbol{k} > 0$ and $\boldsymbol{k} < 0$ indicate the convex and concave points on the contour, respectively. Vector $\boldsymbol{k}$ is sought for its local maxima and minima. Those persisting over scale-space are the most convex and concave corners, respectively. Therefore, the local extrema are back-traced within an arbitrarily chosen neighbourhood applied to $\boldsymbol{k}_\sigma$ over several consecutive blurring steps. A rank of the resulting points is created for each segment and only the top 7% corners are retained. Moreover, corners from all segments are appended to an output list and only 1.5 to 7% of the most convex and concave points are retained per image. Therefore, the count of corners is only about 33% higher than the count of segments.

SUSAN detector [Smith and Brady, 1997] praised for its efficiency is well tailored to detect corners and junctions on segment boundaries. We propose to apply this corner detection approach to region detectors such as segmentations. Figure 2.3 introduces a block diagram of the proposed solution. During the first pass through a segmentation map, a 3×3 mask is applied to detect boundaries between at least two segments and reject uniform areas. Next, a circular mask of an arbitrary radius r_1 is applied and a simple count of the area covered underneath is performed for each segment, respectively.

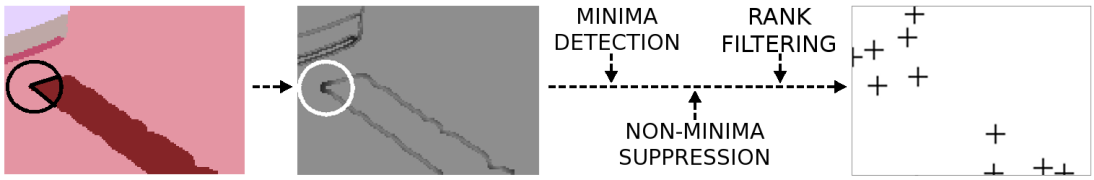


Figure 2.3: Extraction of interest points from segments with SUSAN.

The minimum area amongst segments underneath the circular mask is stored into a new map. If there is no boundary between segments, the default value amounts to $\lceil \pi r_1 \rceil$. For these experiments, $r_1 = 9$ was used. This ensured a good trade-off between the extremely small and large scales of observation. The resulting map is then convolved with a Gaussian kernel of $\sigma = 1.0$ prior to the search for minima, followed by the non-minima suppression using another circular mask of radius $r_2 = 3$. Moreover, the top 7% percent of the most relevant keypoints are retained. This step also constraints the detector to preserve between 4 and 12 corners per segment. As it is demonstrated later, such an approach performs equally well as the heavy duty curvature-based corner detector outlined above. Moreover, this approach is extremely fast due to its simplicity.

2.2.3 Discussion on Boundary and Centre Features

The investigated segmentation methods result in disjoint segments that cover the entire area of images. Thus, the affine regions retrieved by fitting ellipses into the segments provide good coverage of the content in images. By contrast, other interest point detectors often provide many features in some areas and none in others. Moreover, the

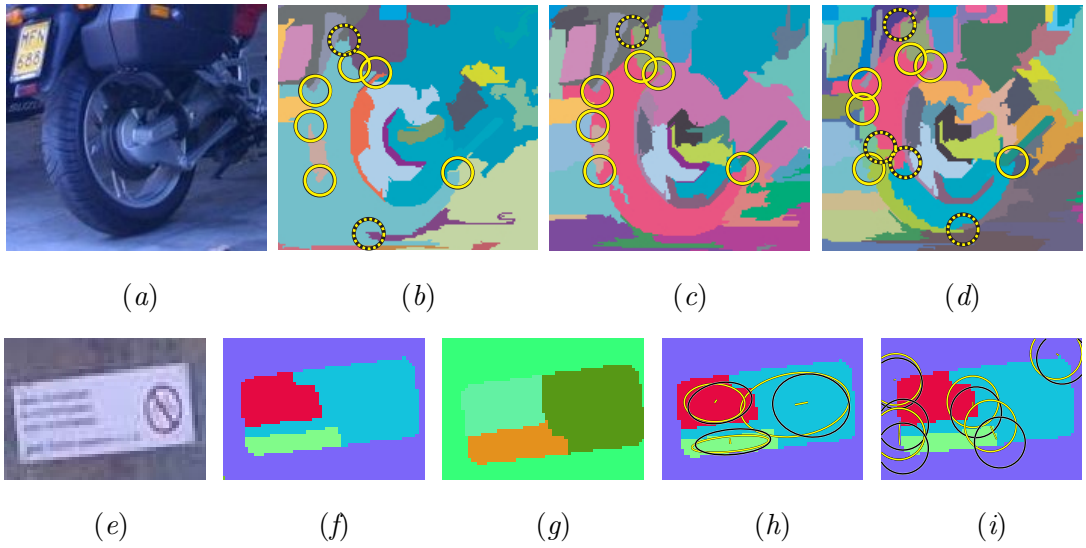


Figure 2.4: (*Top*) An image with (*b*) an under-, (*c*) well-, and (*d*) over-segmented tire and the detected corners. (*Bottom*) The results of segmentation (*f*) before and (*g*) after applying blur, as well as matched (*h*) ellipses and (*i*) corners (see the text).

segment-based features can capture contextually meaningful parts of the objects, *e.g.* cars, wheels, windows, limbs, etc. If the segmentation approaches could produce repetitive results, the features based on such segmentations would also be very repetitive. However, the entire segments tend to suffer from either over- or under-segmentation. Therefore, corners on the boundaries of segments are proposed as more stable features.

Figure 2.4 (top) illustrates a tire of a bike along with the detected corners. Although the tire appears to be segmented out well only in the well-segmented result, there are correctly matched corners (yellow circles) amongst all three segmentations, and only a few of corners remain unmatched (dotted circles). This highlights the repetitive nature of the boundary based keypoints. By contrast, ellipse fitting can be affected by the structural noise that usually occurs at one or more boundaries of a segment.

Figure 2.4 (bottom) shows that given a small blur distortion, two segmentations differ only slightly. Thus, all segments between these two segmentations are matched well by using the ellipses. This is illustrated in plot 2.4 (h) by the overlapping yellow and black ellipses. However, some corners are unmatched in plot 2.4 (i).This is partially due to undetected corners, as well as different matching criteria for both types of such features that will be described next.

To quantify these effects, two complementary measures based on the ground-truth homography H are employed. *The region overlap* proposed in [Mikolajczyk et al., 2005] is defined as the ratio of intersection to union of the reference region R_{μ_r} and the projected region R_{μ_p}:

$$\varepsilon_o = 1 - \frac{|R_{\mu_r} \cap R_{H^T \mu_p H}|}{|R_{\mu_r} \cup R_{H^T \mu_p H}|} \tag{2.3}$$

This measure is used to evaluate the centre-based regions (ellipses) by the percentage of correspondences for which $\varepsilon_o \leq 0.3$. Region R_{μ_r} of segmentation S_1 is said to correspond to region R_{μ_p} of segmentation S_2 (related by H) only if the overlap error criterion is met. *The repeatability* is defined as the ratio of the total number of correspondences to the minimum number of regions $\min(|S_1|, |S_2|)$ shared between the two segmentation maps S_1 and S_2, and related by H.

Alternatively, for the boundary-based points (corners, SUSAN), a criterion based on the distance between an interest point and its nearest projected correspondence is used.

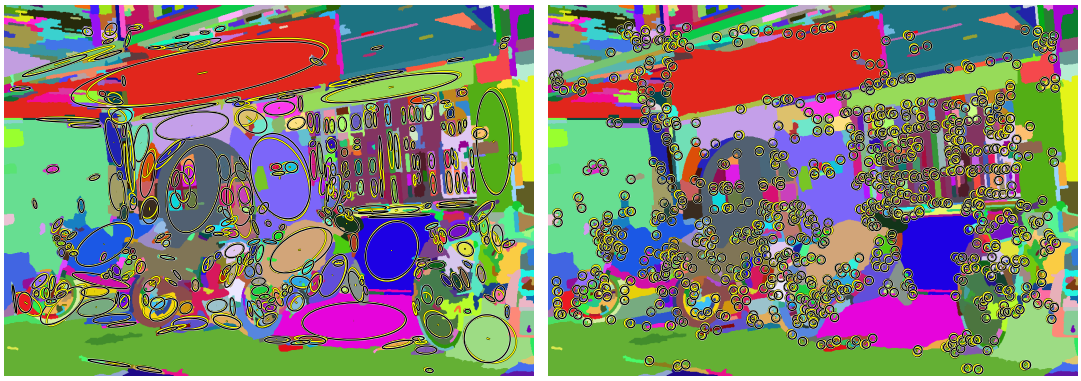


Figure 2.5: Illustration of (*left*) segment- and (*right*) boundary-based features (in yellow) in the reference image together with their correspondences (in black) projected from another image.

The correspondences are considered valid if $\varepsilon_n \leq 4$ pixels. This measure is referred to as Nearest Neighbour (NN).

The overlap based repeatability [Mikolajczyk et al., 2005] helps us examine to what degree the segments of a chosen segmentation approach are preserved over a range of image transformations. The NN repeatability measure [Schmid et al., 2000] is applied to quantify the accuracy of segment boundaries. Figure 2.5 visualises the overlap (left) and the distance-based (right) correspondences.

2.3 Evaluations and Results

This section describes experiments on segmentation-based features. We first discuss our experimental setup and then present the results for the repeatability test, matching of descriptors, complementarity of feature points, and visual categorisation.

2.3.1 Experimental Setup

We exploited a set of well-known test images from [Mikolajczyk et al., 2005]. Each image sequence consists of 6 images with gradually increasing geometric or photometric transformations: bike/blur, boat/scale-rotation, car/illumination, graffiti/affine,

house/JPEG compression, bark/zoom-rotation, tree/blur, wall/affine. These transformations reflect well phenomena taking place during the image acquisition. The availability of the homography ground-truth makes this set useful in such quantitative evaluations. The resolution of images varies from 800×640 to 1000×700 pixels.

According to [Ge et al., 2006], the optimal performance of general-purpose segmentations was achievable if between 10 and 80 segments were produced for 200×200 pixel images. However, it is unclear how segmentations can be compared provided a wide range of their tweaking parameters. As the scale and numbers of objects are not fixed across images, enforcing the arbitrary number of segments does not guarantee they will delineate objects well. To address this issue, we adopted simple heuristics which use EGO to generate three different control sets of segmentation maps at different scales of observation, namely: over-, well-, and under-segmented. The remaining segmentation methods were adjusted to fit to the control sets to their best abilities. In order to avoid damaging effect of exact fitting, we built the histograms of segment sizes for all tested methods and all images from the control sets. The segmentation parameters which produced the most similar histograms to the control set according to χ^2 distance were selected. Finally, we used three sets of parameters for each method. Figure 2.1 (top) illustrates the results of EGO with the under-, well-, and over-segmented maps in plots (c-e). Figure 2.1 (middle, bottom) shows all four segmentation methods on the well-segmented set. A subset of the results is reported in this study. However, the observations and conclusions are drawn from all results unless stated otherwise.

We follow the protocol from [Mikolajczyk et al., 2005] to evaluate the segment features using the repeatability measures, as discussed in section 2.2.3. The results for the state-of-the-art MSER and Hessian detectors operating at the fine scale were added as a reference. We also report the percentage of correct matches obtained with SIFT [Lowe, 1999] to evaluate the proposed features for their applicability in matching.

We additionally investigate *the intra-detector complementarity*. The correspondence sets (repeatable points) for the methods under scrutiny were computed between testing images $1-2$, $1-3$, ..., $1-6$. Further, the correspondence sets of the reference MSER/Hessian detectors were extracted in the same manner. Subsequently, the cor-

respondences from the testing sets having a significant overlap/NN proximity with the correspondences in the referencing sets were removed. The ratio of the remaining correspondences to their original count is called *Exact Complementarity*. If a tested detector yields *e.g.* 90% in such a test, this indicates that the 90% of all repeatable points are novel. The remaining 10% are repeatable but also present in a reference method.

Another measure called *Relaxed Complementarity* differs in a way that the keypoints directly detected by a reference detector on images 2, 3, ..., 6 are used instead of the correspondences from the reference sets 1–2, 1–3, ..., 1–6 when subtracting them from the correspondences for the testing sets. Therefore, the keypoints which are detected as novel with this measure are not implied as definitely repeatable.

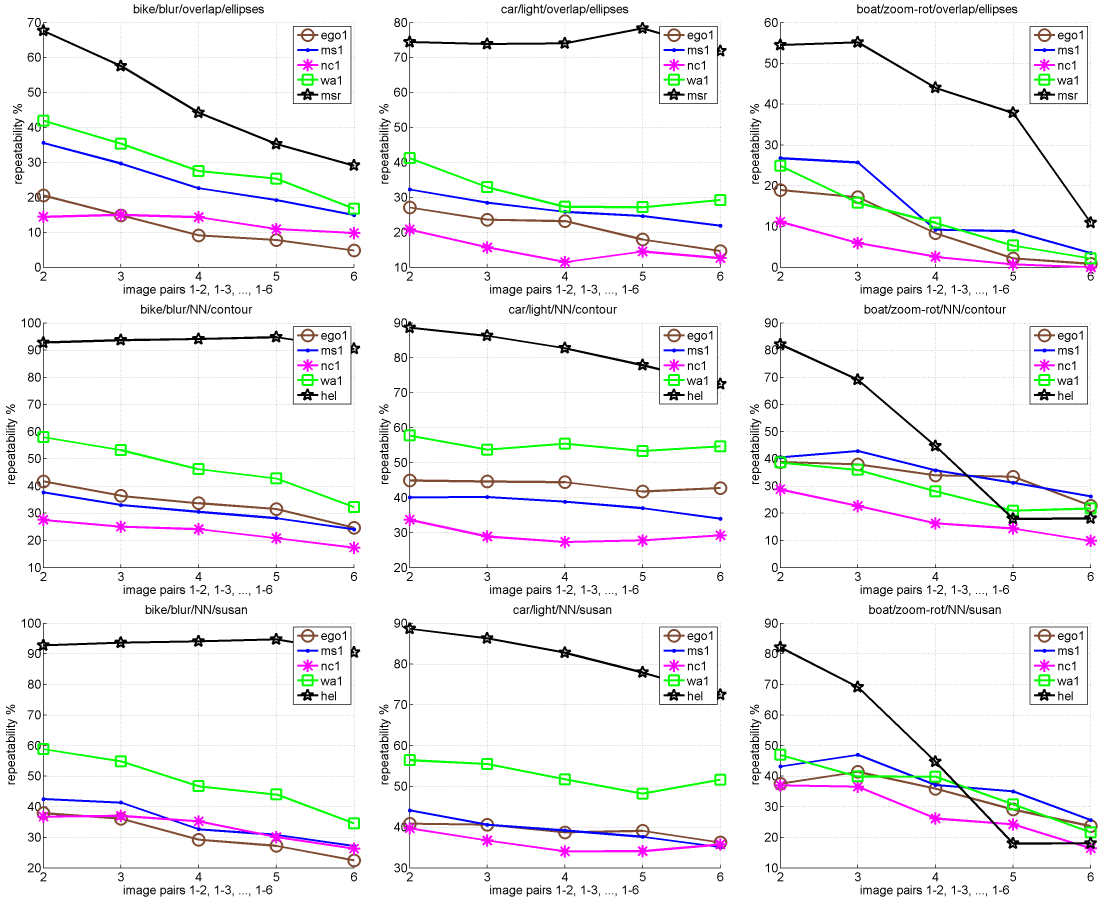


Figure 2.6: The repeatability results for (*left*) bike, (*middle*) car, and (*right*) boat with (*top*) the ellipse-based regions, (*middle*) the curvature-based corners, and (*bottom*) SUSAN corners on the over-segmented set.

Finally, we performed a recognition experiment on the PascalVOC08 dataset [Everingham et al., 2008] to show the classification performance for visual categorisation.

2.3.2 Repeatability of Segmentation Methods

The repeatability of the segment-based features between the original and subsequently distorted images is presented in figure 2.6 for the ellipses (top), the curvature-based based corners (middle), and the SUSAN corners (bottom). The repeatability of the MSER detector was amongst the highest. However, unlike MSER, the proposed approaches do not apply any selection of the most stable regions. A similar observation is valid for the Hessian keypoints compared to the segment-based boundary features. The WA segmentation performed consistently better than the remaining segmentations.

For the over-segmented set, WA was the winner with the repeatability of 42% for graffiti, bike, car, and house. The second best was MS with 30% for bark, boat, tree, and wall. Noticeably, WA behaved better on the structured scenes whilst MS was the second best method scoring on average 33% repeatability. Furthermore, MS was the clear winner for the natural scenes where WA scored rather low. EGO was the third best method reaching roughly 23% for the structured and 20% for the natural sceneries. NC scored 16% on average in all sequences.

For the well-segmented set, WA was comparable to MS on the structured images with about 40% repeatability. MS again outperformed the other methods in the natural scenes with the average of 32%. EGO yielded roughly 20% and NC only 16% across all image categories. In terms of the number of correspondences on the structured scenes, WA produced approximately 150 correspondences between the original and first distorted image, and MS gave 190 correspondences. These numbers reached 200 for MS and 50 for WA on the natural scenes. Regarding the under-segmented set, MS outperformed the other segmentations. For the natural scenes, all segmentations except of EGO produced very few correspondences ($\ll 50$). Therefore, only small persistent object structures were matched. A satisfactory amount of correspondences (≥ 50) was produced for the structured scenes. WA provided the best results for most of the sequences, followed by MS and EGO.

The SUSAN corners proved repetitive in figure 2.6 (bottom), although they performed significantly lower than the Hessian detector (HE). WA won again on the over-segmented set (the structured scenes) with the maximum 58% repeatability. MS led in the natural scenes with the average repeatability of 41% where NC performed second best. WA kept up the same trend for the well-segmented structured scenes with the average repeatability of 54%. MS won consistently in all natural scenes reaching 41%. In case of the over-segmented image set, roughly the same results were obtained for NC and EGO. For the under-segmented set, the structured scenes processed by WA gave again the best average repeatability of 52%. The biggest shift took place on the natural under-segmented images where both EGO and NC were the winners with the similar performance of 35% repeatability. They delivered around 500 and 100 correspondences.

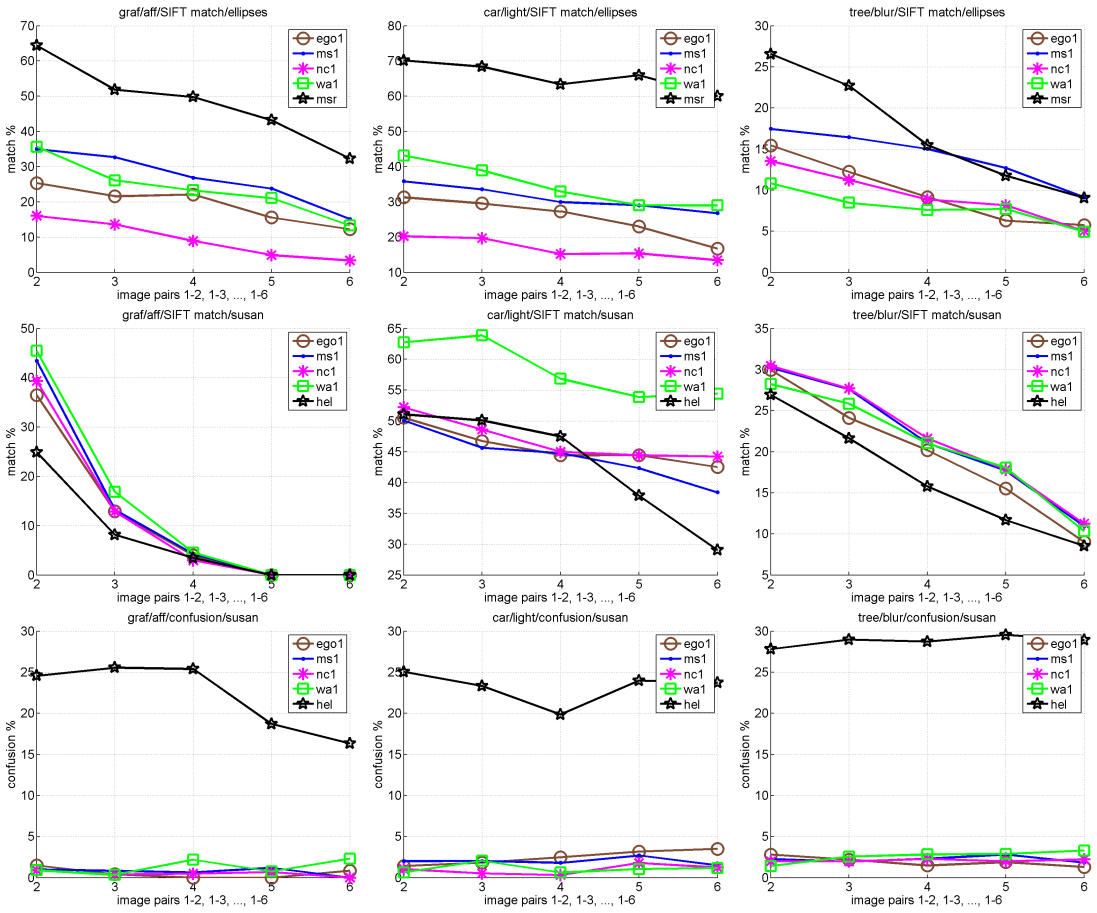


Figure 2.7: The matching results on the over-segmented set for (*top*) ellipses, (*middle*) SUSAN corners. The confusion (*bottom*) for (*left*) graffiti, (*middle*) car, and (*right*) tree.

Concluding, the consistently best performer for the structured categories was WA followed by MS which gave more stable results for the natural scenes. EGO performed on average as the third best method for either scene type. The whole segments were less repetitive than the boundary points due to the frequent spills of segments. WA and MS upheld their stability for both the area- and boundary-based interest points. NC segments produced less stable features. Moreover, figure 2.6 (middle, bottom) shows that the curvature-based and SUSAN corner detectors produce the similar results.

2.3.3 Matching with SIFT

This section provides details on matching with the segmentation-based keypoints combined with SIFT. Both region- and corner-based features are evaluated.

The results for the region-based interest points are displayed in figure 2.7 (top). The attained scores are consistent with the repeatability test from section 2.3.2. Despite the large performance gap if compared to MSER, these regions provide useful features which are unique (WA and MS for the structured and natural scenes, respectively). The radii of the fitted ellipses were increased by a factor of 3 to include the region boundaries into the descriptors and made their sizes comparable to the MSER features. To conclude, the discrepancy between detectors with an embedded stability criterion such as MSER and segmentations suffering from the structural noise is apparent. However, we observed that the stability criterion also suppresses potentially informative keypoints.

Matching with the SUSAN detector brought prime results presented in figure 2.7 (middle). WA outperformed HE by 15%, 20%, 22%, and 7%, for the car, graffiti, boat, and bark sequences respectively. This is in contrast to the repeatability results in section 2.3.2 which showed HE as more repeatable than any combination of SUSAN with the tested segmentations. For the descriptor based matching, SUSAN combined with either of the segmentations outperformed HE for the graffiti, bark, tree, and the wall. Similar trends emerged through other scales of observation. Note that the advantage of MS over WA became clear in the natural scenes. We performed additional experiments to clarify the inconsistency between the repeatability and the matching scores for HE.

Figure 2.7 (bottom) gives us an insight into how many points from a given image were

matched with more than one point in the corresponding transformed image. HE produced many multi-matches for the same local structures in contrast to the segmentation based points. This indicates much higher redundancy of the HE features. Also, we attribute good performance of SUSAN to the segmentations which aim to capture the entire distinct regions. We argue that these results are also due to the segmentation-based corners which are very salient keypoints as they occur on the perimeter of two or more areas considered dissimilar by a given segmentation. Therefore, descriptors are rarely extracted from the visually uniform uninformative parts of images.

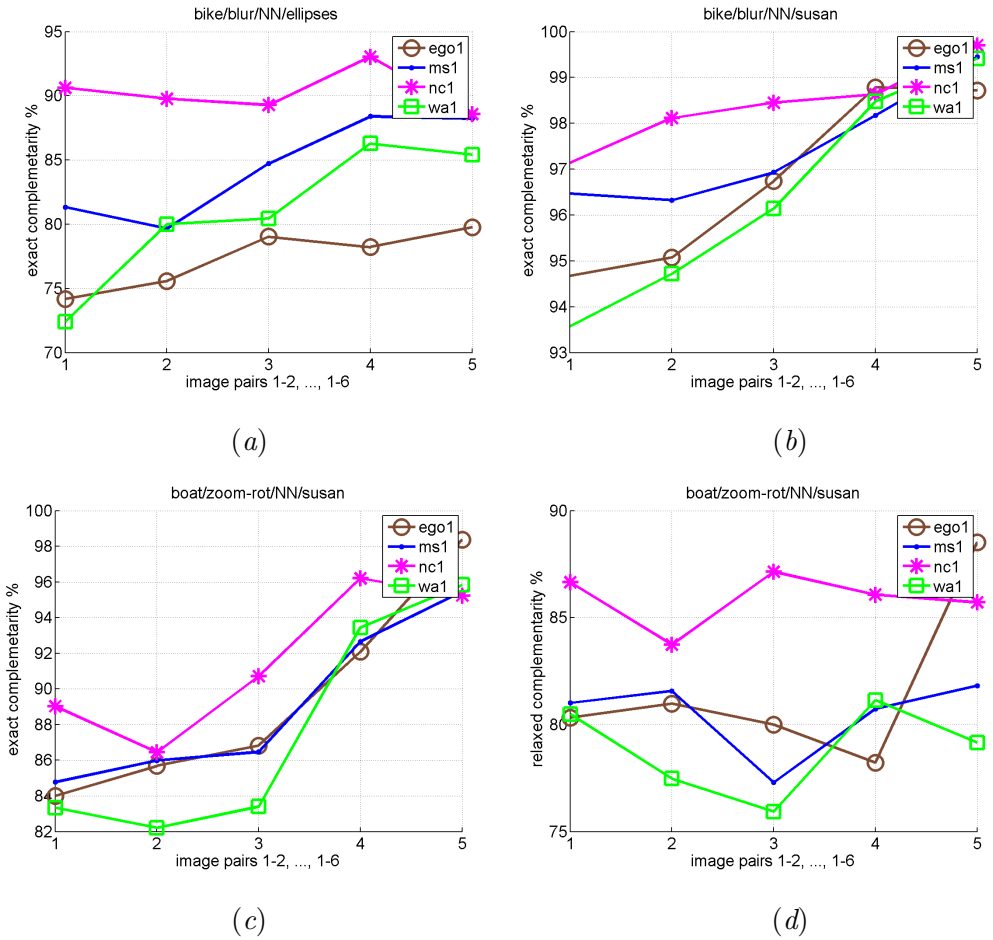


Figure 2.8: The inter-detector complementarity of the region-based/SUSAN detectors with MSER/Hessian (the ratio of the novel repeatable to all repeatable points). Bike/EC for (*a*) ellipses and (*b*) SUSAN. Boat/SUSAN given (*c*) EC and (*d*) RC.

2.3.4 Inter-detector Repeatability

The following experiment provides details on the level of complementarity amongst the examined region- and corner-based features from the segmentation maps compared to the MSER/HE reference detectors, respectively. Highly complementary detectors can be used together to improve performance of matching or recognition. Figure 2.8 presents the obtained complementarity results. The higher the Exact Complementarity (EC) measure the more novel repeatable interest points are detected with respect to the reference methods. For the region-based keypoints extracted from the well-segmented images, EC amounted to 79%, 78%, 93%, and 83% for EGO, MS, NC, and WA respectively. These are the average values concerning the testing image pairs $1-2$. The following consecutive testing pairs $1-3$, ..., $1-6$ yielded mostly monotonically increasing scores. The stronger were the image distortions the more novel keypoints were observed, although at a cost of fewer correspondences. The corner-based feature points yielded the following scores: 91%, 92%, 93%, and 90% respectively. The Relaxed Complementarity (RC) measure resulted in similar trends which were consistently lower by between 2 and 4% on average. The RC measure suffers from a bias towards the non-repeatable noise detected by a reference detector. Figure 2.8 (a, b) shows EC for bike given the segment-based regions and the SUSAN corners, respectively. Figure 2.8 (c, d) shows the exact and relaxed scores for boat. RC usually increases with EC. However, the reference noise can affect these scores as shown in plot (d).

The complementarity score is expected to remain below 100%, as the most distinctive features within an image should be extracted by any sort of a robust detector. To conclude, the segmentation-based methods introduce a significant number of the additional complementary features to the state-of-the art interest point detectors.

2.3.5 Visual Object Category Recognition

In this experiment, the PascalVOC08 set [Everingham et al., 2008] was used to compare the proposed corner features to MSER/Hessian points, all combined with the SIFT descriptor. We applied *Pyramid Match Kernel* (PMK) approach with SVM from [Grauman and Darrell, 2005] with 4 pyramid levels and the branch factor equal 20.

features	dense	HE	MSER	MS	WA	WS	WA+D
#regions per img	-	1710	1677	1674	**1609**	1785	-
MAP (%)	-	30.78	31.37	32.51	**33.76**	31.78	-
#regions per img	**3690**	2417	3886	3877	**2905**	2796	4108
MAP (%)	**33.77**	31.49	33.00	34.50	**36.01**	33.14	34.70

Table 2.1: The MAP results for the PascalVOC08 dataset.

The Pyramid Match Kernel (PMK) scheme was trained for 20 object classes on the training set consisting of 2111 images, and tested on the validation set of 2221 images. As a reference approach, we applied dense feature sampling on a regular grid with the sampling interval of 8, 14, 20, and 26 pixels. This gave 3690 features on average per image. The SIFT descriptors for both reference and segmentation-based interest points were generated with patch radii of 16, 24, 32, and 40 pixels. Forcing the fourfold scales upon MSER/HE also resulted in their best performance compared to the affine/scale invariant configuration. The MSER, HE, MS, and WA corner features were tested for two different numbers of descriptors per image. In addition, we show results for the Watershed based detector without the anisotropic filter (WS) to demonstrate its advantage. Table 2.1 shows the MAP scores computed over all 20 object categories. The experiments performed in [Nowak et al., 2006] explain the poor performance of MSER/HE in visual categorisation. WA gave the highest scores of 36.01% MAP. It required 1.3× less features than in the case of dense sampling strategy (33.77% MAP). With 2.3× less features, WA was still on a par with the reference approach. This clearly demonstrates the saliency of the segmentation-based features in contrasts with [Nowak et al., 2006]. To explain this, we combined the WA keypoints with dense sampling (WA+D). The uniform image regions that resulted in large segments were not represented by the WA keypoints. Therefore, only these regions were supplemented by the dense sampling keypoints. This resulted in a 1.3% drop in MAP. We conclude that oversampling the uniform image regions can be detrimental to visual categorisation.

This experiment validates our observations from section 2.3.3 on a larger dataset. Note that the obtained results are not directly comparable with the top scores for in the literature as we used only one kernel and the validation data set for testing.

2.4 Conclusions

The performed experiments investigate the features for matching and recognition which were extracted from the segmentation maps. The best performer for the structured scenes was WA while MS was second best in this category and the best for the natural images (frequent textures). These two methods were followed by EGO yielding slightly lower repeatability scores. The region-based interest points proved fairly stable, though such detectors would benefit from a selection scheme based on some stability measure similar to the one applied in MSER.

The junctions of segments were proved as very stable features with means of SUSAN. Even in case of poor region-related performance (*e.g.* under-segmentation), EGO still yielded good results when matching with SIFT. Again, WA and MS turned out to be the two most stable segmentations. It emerged that the interest points based on strong curvature of boundaries between regions are more suitable for both matching and recognition than simple blob-based features. It seems that the repeatability, matching and recognition benefit from the well- and over-segmentation strategies since they produce higher numbers of stable features. Although using the information carried by the segmentation maps may seem a daunting task due to their structural noise, they were demonstrated to benefit visual categorisation by focusing on salient image structures and suppressing keypoints from uniform regions.

Chapter 3

Segmentation Based Image Descriptors

This chapter investigates the segmentation-based image descriptors for object category recognition. In contrast to the commonly used descriptors computed over regions provided by interest point detectors, the proposed descriptors are extracted from pairs of adjacent regions given by a segmentation method. In this way, we exploit semi-local structural information from images. We propose to use the segments as spatial bins for descriptors of various image statistics based on gradient, colour, and affine shape of regions. The proposed descriptors are evaluated on the standard recognition benchmarks.

3.1 Introduction

Adequate image representations have been shown as crucial for the performance of image retrieval and recognition systems. State-of-the-art systems rely on the interest point detectors such as MSER, Hessian, and Harris [Mikolajczyk et al., 2005] typically combined with the local image descriptors, *e.g.* SIFT [Lowe, 1999]. For visual categorisation, dense sampling has been advocated over the keypoint extraction [Everingham et al., 2007]. Chapter 2 showed that the unsupervised segmentation maps constitute a good alternative to both standard keypoint detectors and dense sampling

strategies. With less interest points derived from these maps, they outperform the dense sampling approach which typically scores the highest in the challenging Visual Object Category Recognition [Everingham et al., 2007]. This is due to the saliency of the detected curvature-based keypoints between the segment boundaries, as well as due to a full coverage of images with the extracted segments. This is in contrast to the sparsely distributed interest points. This chapter investigates direct applicability of the segmentation maps to devising image representations that cover all regions of the processed images. Such an approach makes use of the semi-local structures formed by the segments. In order to capture the boundaries of the objects, as well as the gradient within their areas, the adjacent pairs of segments are processed as spatial support for extracting various measurements from images. We argue that these pairs form good spatial hypotheses which capture an essential gradient-based appearance of an object. Furthermore, multiple segmentation maps extracted with different parameters enrich such a hypothesis space. To our best knowledge, there are no other methods that use segmentations as the spatial hypotheses for shape of the multiple descriptor cells.

3.1.1 Related work

Segmentation maps have been used widely as an auxiliary grouping cue instead of the common bounding boxes [Malisiewicz and Efros, 2007]. It was also shown in [Ott and M.Everingham, 2009] that enhancing foreground/background hypotheses improves the classification results. Furthermore, extremal curvatures of segments were found to serve well as the salient points outperforming the dense sampling strategies in chapter 2. An optimal spatial arrangement of the descriptor bins has been recently investigated in DAISY [Tola et al., 2008] which is aimed for dense matching. DAISY comprises several circular regions which are arranged in a polar manner resembling petals of a flower. Learning local image descriptors [Winder and Brown, 2007] can be performed by selecting several operations: type of a gathered histogram evidence, shape of the spatial bins. A blob-based representation proposed in [Carson et al., 1999], where a small number of segments corresponding to the entire objects are described by colour and texture, is somewhat similar in spirit to work presented in this chapter.

Figure 3.1: Segmentations at the several scales of observation (see text for details).

3.2 Proposed Image Descriptors

Segmentation maps act as the spatial hypotheses to delineate distinct parts of objects. Multiple measurements can be taken from images within such defined areas. In natural images, objects appear at many different scales. Therefore, segmentation maps at multiple scales of observation are extracted and used to build more accurate object representations. For this purpose, the Watershed segmentation proposed in chapter 2 is employed. The average numbers of segments in the image were varied by factor of 1.6× between 4 consecutively coarser scales of observation S_0, ..., S_3 presented in figure 3.1 from top left towards bottom right.

3.2.1 Spatial Arrangement

To establish a baseline system, we devised a basic descriptor such that each segment corresponded to one descriptor vector (a single spatial bin). The statistics of orientations of image gradients were extracted within areas of segments including boundaries to form 12 dimensional vectors (we refer to it as **V0**).

Moreover, in order to exploit the semi-local image structures in the form of spatial arrangements of segments, all pairs of the adjacent segments per image were used to

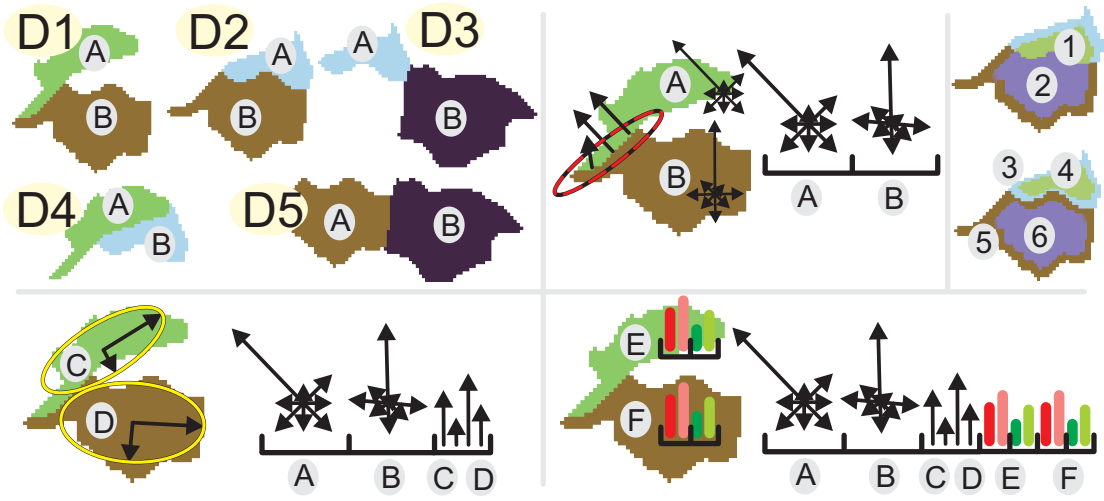


Figure 3.2: The architecture of the proposed descriptors (see text for details).

build descriptors, each comprising two spatial bins. Figure 3.2 (top left) illustrates how segments corresponding to the jockey's head from figure 3.1 (bottom right) form pairwise combinations yielding vectors $\boldsymbol{D}_1$, ..., $\boldsymbol{D}_5$, ..., $\boldsymbol{D}_N$. The repeatability of such descriptors can be ensured by preserving of the order of the spatial bins in which they were extracted from images. Therefore, the segments are always grouped from top to bottom and from left to right. The structural noise and image distortions may affect the order in which segments are extracted. However, with multiple segmentation hypotheses and several training images per class, the proposed strategy works well.

Figure 3.2 (top middle) illustrates how the pairwise statistics from regions A and B (note the order) are gathered to form a descriptor referred to as **V1**. The number of the orientation bins defined on each spatial bin amounted to 8, 10, or 12 per experiment.

Moreover, various combinations can be formed from the pairs of segments. We investigated if including regions around the boundaries of a segment pair independently from their interiors can further improve these representations. Therefore, vectors formed from the regions 3 and 5 in figure 3.2 (top right) were tested (descriptor variant **V2**).

To measure the levels of discriminative information within the segment interiors only, a descriptor (variant **V3**) was designed from regions 4 and 6 in figure 3.2 (top right).

The statistics gathered only within small margins along boundaries of a joint segment $A \cup B$ capture primarily the edge between regions. Therefore, influence of the strong

Figure 3.3: Dominant orientations and sizes of segments can be repeatable.

gradients along boundaries of $A \cup B$, except for their common boundary, is decreased. Hence, we combined regions 1 and 2 only. This is the descriptor variant **V4**.

Lastly, we attempted to answer if boundaries and interiors of segments convey a complementary information. Therefore, regions 3, 4, 5, and 6 were arranged into four spatial bins forming descriptor. This descriptor variant is called as **V5**.

3.2.2 Capturing Shape of Segments

The shape of segments is captured by the orientations of image gradients, in particular from segment boundaries. However, the dominant shapes of segments as well as their relation can be additionally encoded by the eigenvectors of entire segments. Figure 3.3 shows three segmentations performed on images containing birds. Both dominant orientations and sizes of segments seem to repeat. Therefore, such representations are worth capturing. Figure 3.2 (bottom left) shows that ellipses are fitted into the adjacent segments to capture their dominant axes. Extracted eigenvectors and eigenvalues provided auxiliary descriptor coefficients. Three scenarios were investigated:

- The 4, 6, or 8 orientation bins are addressed by the angles $\phi_k = \phi(\boldsymbol{e}_k)$ of eigenvectors $\boldsymbol{e}_k$ given a spatial bin. Each angle is quantised to choose one of the orientation bins. The bin is then incremented by eigenvalue $e_k = \| \boldsymbol{e}_k \|_2$.
- The 2 bins conveying phase values ϕ_1 and ϕ_3 for two spatial bins.
- The 4 bins consisting of eigenvalues $e_1, ..., e_4$ for two spatial bins.

3.2.3 Colour Statistics

Colour stimuli provide cues complementary to the orientation-based features, as explained in [van de Sande et al., 2008]. To capture the gist of a semi-local colour profile, low dimensional colour histograms were collected from the image regions indicated by the segments. We experimented with the Opponent and YUV colour spaces. We also investigated the luminance-based additional normalisation on the Opponent colour space. To decide how to quantise the histogram bins, we estimated their distributions on the PascalVOC08 training set [Everingham et al., 2008], and concluded that the marginal distributions of chromaticity components C_1 and C_2 were Laplacian shaped. This suggested that the chromaticity components could be independent. Thus, twofold ideas were examined: 5×5 bins for the joint statistics of C_1 and C_2 per segment, and separate 5 bins for C_1 and 5 bins for C_2 statistics per segment. Note that the Opponent and YUV colour spaces are both light intensity and shift invariant [van de Sande et al., 2008]. These spaces are also similar semantically.

3.2.4 Data Assignment and Normalisation

A bilinear approximation was examined for assigning the data to the spatial bins. For the segmentation-based descriptors, the linear weights depend on the distance from the boundary between segments A and B. Moreover, a bilinear approximation for the orientation bins was investigated. Various measurements are taken within each spatial bin, to wit: the orientations of image gradients, eigenvalues of segments, histograms of colours. Therefore, we experimented with normalising each measurement per spatial bin, as well as per pair of spatial bins. The best results were achieved for each type of information normalised to unit vectors per spatial bin per measurement type, except for the histograms of eigenvalues which were both normalised jointly.

3.3 Evaluations and Results

The initial tests were performed on the PascalVOC08 set [Everingham et al., 2008] while the final tests were carried out on the PascalVOC07 set [Everingham et al., 2007].

3.3.1 Experimental Setup

The PascalVOC08 set consists of 2111 training and 2221 validation images for testing. No testing corpus is available for this dataset. PMK and SVM classifier from [Grauman and Darrell, 2005] were used. The PascalVOC07 set consists of 2501 training, 2510 validation, and 4952 testing images. The χ^2 with RBF kernel (χ^2_{RBF}) and the KDA classifier [Tahir et al., 2009] were employed for this dataset. Both classification systems were trained for the same 20 object classes. For PMK, the same setup as in section 2.3.5 was recreated (4 pyramid levels with the branch factor equal 20). For χ^2, hierarchical k-means clustering with 10×400 clusters and Soft Assignment (SA) [van Gemert et al., 2010] were applied. Furthermore, the dense sampling scheme on a regular grid with the intervals of 8, 14, 20, and 26 pixels was applied to generate the reference SIFT [Lowe, 1999] descriptors with patch radii of 16, 24, 32, and 40 pixels.

The Average Counts of Features. The performed experiments aimed at using low numbers of features. Segmentation scales S_0, ..., S_3 yielded approximately 596, 590, 353, and 199 feature vectors per image. Combined segmentation scales S_{123}, S_{0123}, and S_{01234} produced 1148, 1738, and 2202 vectors. These numbers compare favourably to 3690 densely sampled SIFT descriptors.

3.3.2 Initial Experiments

The initial experiments were carried out on PascalVOC08 as it consists of a fewer number of images. The goal was to compare different approaches for fusing the segment-based statistics. Table 3.1 summarises results in terms of MAP computed over all 20 categories. The experiments were performed for scale S_1 until stated otherwise. The

variant	$V0$	$V1^{Do}_{S}$	$V1^{o8}_{S}$	$V1^{o10}_{S}$	$V1^{o12}_{S}$	$V1_{H}$	$V1_{HSb}$	$V1_{HSbt}$	$V2$
MAP %	23.88	22.6	24.91	26.6	27.43	27.78	28.12	28.45	27.05
V3	V5	$V1^{Eg}_{HSbt}$	$V1^{Eg}_{Op}$	$V1^{Eg}_{\widehat{Op}}$	$V1^{Eg}_{UV}$	$V1^{Eg}_{S_{03}}$	$V1^{Eg}_{OpS_{03}}$	DSIFT	
17.26	28.09	28.65	30.62	29.61	30.67	**32.32**	**34.00**	**33.77**	

Table 3.1: The MAP results for the experiments on the PascalVOC08 set.

final dimensionality for each proposed descriptor variant is indicated in brackets. $V0$ is a single spatial bin descriptor ($12D$) as explained earlier. $V1_S^{Do}$ denotes a pair-of-segments descriptor with 2×12 bins (separately normalised bins, bilinearly approximated) with a built-in orientation invariance based on the dominant orientation mechanism [Lowe, 1999]. Typically, such an invariance decreases performance of PMK. However, other applications may require this type of invariance. $V1_S^{o8}$ to $V1_S^{o12}$ are variants of $V1$ with 2×8, 2×10, and 2×12 orientation bins. According to the results, an increase in the number of orientation bins leads to a slight increase of scores.

$V1_H$ is a hard assigned variant (no bilinear approximation) of $V1$ with 2×12 bins. $V1_{HSb}$ and $V1_{HSbt}$ are the descriptor variants comprising hard assignment and the gradients obtained with the Sobel operator. For $V1_{HSbt}$, the gradient magnitudes below an arbitrarily low threshold were not included into the orientation bins. We note that the hard assignment outperforms the bilinear assignment of the data. We attribute this to the boundaries between pairs of segments to be already strong hypotheses distributing gradients proportionally to the spatial bins. As only two spatial bins are employed, smoothing should be avoided to preserve their distinctiveness.

Furthermore, the alternative spatial arrangements for pairs of segments were investigated. $V2$ to $V4$ comprise 2 while $V5$ have 4 spatial bins. They all use the hard assignment, gradient computed with the Sobel mask, and the noise threshold, as in $V1_{HSbt}$. Removing the interiors of segments did not bring any benefit (case of $V2$). Retaining these interiors while removing the boundaries of segments demonstrates that some information is retained in these interiors, as the results of $V3$ show. This may be due to the blended transition of the boundary edges, as well as the texture. Variant $V4$ focusing on the boundary between segments A and B was a poorer performer than $V2$. Variant $V5$ did not deem descriptors any more informative than ordinary $V1_{HSbt}$.

The remaining experiments in this section were concerned with exploitation of segment shapes, their arrangements, and the colour statistics. We selected the most successful variant $V1_{HSbt}$ and combined it with 3 other variants of eigenvector based representations. Descriptors using the orientations of eigenvectors decreased the results whilst the histograms of eigenvalues ($4D$) combined with $V1_{HSbt}$ improved performance. This

variant	$V1_{HSbt}$	$V1^{Eg}$	$V1^{Eg}_{Op}$	$V1^{Eg}_{OpF}$	$V1^{Eg}_{S_{03}}$	$V1^{Eg}_{OpS_{13}}$	$V1^{Eg}_{OpS_{03}}$
MAP %	39.14	39.73	43.39	43.00	43.44	**45.26**	**46.02**
$V1^{Eg}_{OpS_{04}}$	DSIFT	OSIFT	OS+$V1$	OS+$V1^*$	BK	BK+$V1$	
47.54	**44.81**	**46.56**	**53.81**	**57.8**	**61.82**	**63.34**	

Table 3.2: The MAP results for the experiments on the PascalVOC07 set.

variant is referred to as $V1^{Eg}_{HSbt}$ ($28D$). For clarity, let us drop the subscript and call the most successful variant as $V1^{Eg}$. Its extensions with $2\times2\times5$ bins of the Opponent colour statistics are denoted as $V1^{Eg}_{Op}$ ($48D$), luminance-normalised Opponent colour statistics as $V1^{Eg}_{\widehat{Op}}$ ($48D$), and YUV statistics as $V1^{Eg}_{UV}$ ($48D$). The best results were delivered by $V1^{Eg}_{UV}$ and $V1^{Eg}_{Op}$. Finally, to benefit from the multiple segmentations, multiple feature vectors were appended across scales S_0, ..., S_3 to form $V1^{Eg}_{S_{03}}$ ($28D$) and $V1^{Eg}_{OpS_{03}}$ ($48D$) descriptor variants. For convenience, the results for these variants are indicated in green in table 3.1. With 5.6× less data, the latter variant outperformed the dense SIFT descriptor (DSIFT).

3.3.3 Final Evaluations

Having identified the best configuration of the proposed descriptor, further tests were performed on the PascalVOC07 set. This section also evaluates how complementary are the proposed descriptors to SIFT. For this purpose, χ^2 kernels built from the proposed descriptor and the state-of-the-art kernels from [Tahir et al., 2009] were combined together. Table 3.2 presents the results for both single kernels and the most interesting fusions. As previously, $V1^{Eg}$ seemed to score a bit higher than $V1_{HSbt}$. This confirms that the $4D$ histogram of eigenvalues improves the representations. Given that the Opponent and YUV colour spaces are closely related, we report only results for $V1^{Eg}_{Op}$ ($48D$) and $V1^{Eg}_{OpF}$ which extends $V1^{Eg}$ with $2\times5\times5$ colour bins ($78D$). In spite of the higher dimensionality of such distributions, no additional information was captured. This can be explained by resemblance of these distributions to the product of the marginal colour distributions.

To achieve invariance to the scale changes, segmentations were extracted at multiple scales of observation. $V1^{Eg}_{OpS_{13}}$ is a collection of $V1^{Eg}_{Op}$ across scales S_1, ..., S_3. It per-

formed on a par with the dense SIFT descriptor (DSIFT) given 8.6× less data. $V1^{Eg}_{OpS_{03}}$ turned out to be an even better descriptor representation. $V1^{Eg}_{OpS_{04}}$ outperformed the dense Opponent SIFT descriptor (OSIFT) from [van de Sande et al., 2008] with 13.4× less data. For convenience, the results for these variants are indicated in green in table 3.2. OS+V1 denote a kernel fusion of the Opponent SIFT and the best segmentation descriptors. Despite both descriptors employ the colour statistics, their combination resulted in a significant gain in performance. OS+$V1^*$ are the results for $V1$ merged with the spatial version of kernel OS [van de Sande et al., 2008]. This improves the results by 4%. Moreover, BK is a range of kernels built from several state-of-the-art descriptors [Tahir et al., 2009]. BK+$V1$ represents their fusion with our kernel based on $V1^{Eg}_{OpS_{04}}$, with a further improvement of 5.5%. For convenience, the best results for the multiple kernel fusions are indicated in red in table 3.2.

3.4 Conclusions

The experiments proved that the segmentation-based image descriptors are highly informative, competitive, and complementary to SIFT. A computational advantage of such representation was noticeable during clustering with k-means. Reduced dimensionality and small numbers of descriptors resulted in faster computations. Unsupervised segmentations delivered good spatial hypotheses that split objects into descriptive semi-local regions at multiple scales of observation. Furthermore, such representations resulted in a thorough coverage of images with descriptors, as opposed to the sparse interest points. Moreover, the visually uniform regions were often delineated as sole segments in segmentation maps. Therefore, this helped to reduce their contributions in the final image representations. The results show that the proposed representations outperform the state-of-the-art reference descriptors with 5.6× less data and achieve comparable results to them with 8.6× less data. The proposed descriptors are complementary to SIFT and achieve state-of-the-art results when combined together within a kernel based classifier. With 63.34%, the final kernel BK+$V1$ outperformed a state-of-the-art approach from [Yang et al., 2012a] which scored 62.2%.

Chapter 4

Reconstruction Error in Soft Assignment

Visual Word Uncertainty, also known as Soft Assignment (SA), is a well established technique for the BoW model that transforms local image descriptors into histograms. This is accomplished by a flexible assignment of the descriptors to a visual vocabulary. Recently, a substantial improvement in visual categorisation has been achieved with Linear Coordinate Coding (LCC). This chapter investigates the SA model. Specifically, it is shown that SA, a model derived from Gaussian Mixture Model (GMM), can act as an approximation to the LCC model. This is achieved by an optimisation of the so-called *smoothing factor* of SA. Such an optimisation combines SA with the quantisation loss used by LCC. Minimising the quantisation loss in this manner correlates well with the best classification performance, which is demonstrated on two popular datasets and various image descriptors. Specifically, SIFT and the segmentation-based semi-local descriptors presented in chapter 3 are employed.

4.1 Introduction

Transforming the local image descriptors into the image signatures lies at the heart of the BoW model. The search for appropriate coding schemes expressing robustly the content of images has been a subject of recent activity in the community. A number

of methods have been proposed including Hard Assignment (HA), SA [van Gemert et al., 2010], and a family of the LCC methods [Yu et al., 2009]. They entail Sparse Coding (SC) [Yang et al., 2009], Locality-constrained Linear Coding (LLC) [Wang et al., 2010], and other methods.

HA associates each descriptor vector with the nearest visual word of a k-means dictionary. Whilst this provides with a fair expressive power, this model has poor quantisation properties, *e.g.* a descriptor on the cluster boundary may be assigned to one or another word due to the stochastic noise. SA mitigates such an effect by employing soft contributions of each descriptor to its closest visual words in the dictionary. This was initially implemented as heuristics such as assigning a given descriptor to the k-nearest visual words. Subsequently, SA [van Gemert et al., 2010] was found to be a more appropriate assignment scheme. However, it requires tiresome cross-validation to determine the so-called smoothing factor that has impact on the classification performance of this model. Moreover, to improve the quantisation properties of the assignment schemes, the LCC coding was proposed [Yu et al., 2009]. This method expresses each descriptor vector as a linear sparse combination of neighbouring dictionary anchors. The ℓ_1 norm regularisation computed over the resulting assignments favours only a small subset of non-zero assignment coefficients, this leads to the so-called *sparsity*. Moreover, SC combined with SPM produced very promising results in [Yang et al., 2009].

This chapter is concerned with bridging the gap in understanding of SA in the context of LCC, as the first approach can be viewed as an approximation of the latter one. Moreover, SA is also shown to be related to Component Membership Probabilities of GMM [Bilmes, 1997]. Foundations of LCC are exploited to find the optimal smoothing factor for SA by optimising the quantisation loss that is typically used by the LLC family. Minimising the proposed cost function is shown to correlate well with the best classification performance.

4.2 Derivation of Soft Assignment

Given a mixture of K Gaussian functions G with the parameters $\theta = (\theta_1, ..., \theta_K) = ((w_1, \boldsymbol{m}_1, \boldsymbol{\sigma}_1), ..., (w_K, \boldsymbol{m}_K, \boldsymbol{\sigma}_K))$, the density estimation problem can be solved by op-

timising the following cost w.r.t. θ:

$$\Lambda(\mathcal{X};\theta) = \prod_{n=1}^{N}\sum_{k=1}^{K} w_k G(\boldsymbol{x}_n;\boldsymbol{m}_k,\boldsymbol{\sigma}_k) \tag{4.1}$$

K denotes the number of components, component index is indicated by $k=1,...,K$, w_k are the component mixing probabilities, $\boldsymbol{m}_k$ are the Gaussian means, $\boldsymbol{\sigma}_k$ are the deviations, $\boldsymbol{x}_n$ are the descriptors from a given dataset such that $n=1,...,N$. The *membership probability* of component k being induced by descriptor $\boldsymbol{x}$ is:

$$p(k|\boldsymbol{x}) = \frac{w_k G(\boldsymbol{x};\boldsymbol{m}_k,\boldsymbol{\sigma}_k)}{\sum_{k'=1}^{K} w_{k'} G(\boldsymbol{x};\boldsymbol{m}_{k'},\boldsymbol{\sigma}_{k'})} \tag{4.2}$$

Note that the parameters of the model in equation (4.1) have a vast number of degrees of freedom and therefore are further reduced to $\theta=(\theta_1,...,\theta_K)=((\boldsymbol{m}_1,\sigma),...,(\boldsymbol{m}_K,\sigma))$ by fixing all mixing probabilities $w_1=w_2=...=w_K\neq 0$ to be equal and having a single σ parameter such that $\boldsymbol{\sigma}_1=\boldsymbol{\sigma}_2=...=\boldsymbol{\sigma}_K=\sigma\neq 0$. Therefore, the cost function can be rewritten as:

$$\Lambda(\mathcal{X};\theta) = \prod_{n=1}^{N}\sum_{k=1}^{K} G(\boldsymbol{x}_n;\boldsymbol{m}_k,\sigma) \tag{4.3}$$

This leads to the expression for the membership probability of component k being selected given descriptor $\boldsymbol{x}$:

$$p(k|\boldsymbol{x}) = \frac{G(\boldsymbol{x};\boldsymbol{m}_k,\sigma)}{\sum_{k'=1}^{K} G(\boldsymbol{x};\boldsymbol{m}_{k'},\sigma)} \tag{4.4}$$

Note that the above expression is also used by the SA model. Its role is to assign descriptor $\boldsymbol{x}$ to the visual vocabulary. To compute the k^{th} entry to the final histogram representing a given image, an expected value of $p(k|\boldsymbol{x}_n)$ is computed over descriptors $\boldsymbol{x}_n$ from that image, where n indicates each descriptor. Note that σ could be estimated by minimising the GMM density in equation (4.3). However, σ estimated in such a way proved underestimated as the density estimation and coding are different problems.

4.3 Combining Soft Assignment and Linear Coordinate Coding

The foundations of LCC are provided in [Yu et al., 2009]. We discuss only the formulations essential to the work in this chapter. *Coordinate Coding* is a pair $(f,\mathcal{M})$,

where $\mathcal{M} \in \mathbb{R}^{D \times K}$ is a visual dictionary and f is a mapping of a descriptor $\boldsymbol{x} \in \mathbb{R}^D$ to an image signature represented by vector $\boldsymbol{\phi} = [f_{\boldsymbol{m}}(\boldsymbol{x})]_{\boldsymbol{m} \in \mathcal{M}} \in \mathbb{R}^K$. One can further impose that $\sum_{\boldsymbol{m}} f_{\boldsymbol{m}}(\boldsymbol{x}) = 1$ and $f_{\boldsymbol{m}}(\boldsymbol{x}) \geq 0$ if histograms are required. The inverse of mapping f can be expressed as $\boldsymbol{x} = f^{-1}(\boldsymbol{\phi}, \mathcal{M})$. The LCC methods approximate the inverse by the linear combination $\hat{\boldsymbol{x}} = \sum_{\boldsymbol{m} \in \mathcal{M}} f_{\boldsymbol{m}}(\boldsymbol{x})\boldsymbol{m}$. Thus, the residual error of the approximation of descriptor vector $\boldsymbol{x}$ becomes:

$$\xi^2(\boldsymbol{x}) = \left\| \boldsymbol{x} - \sum_{\boldsymbol{m} \in \mathcal{M}} f_{\boldsymbol{m}}(\boldsymbol{x})\boldsymbol{m} \right\|_2^2 \tag{4.5}$$

Moreover, the approximation error of all descriptors can be expressed as the expected value of terms $\xi^2(\boldsymbol{x}_n)$ over all descriptor indexes $n = 1, ..., N$, or simply as a sum $\xi^2 = \sum_n \xi^2(\boldsymbol{x}_n)$. Such a defined error is equivalent to the quantisation error also known as the quantisation loss [Yu et al., 2009]. Therefore, combining equation (4.4)

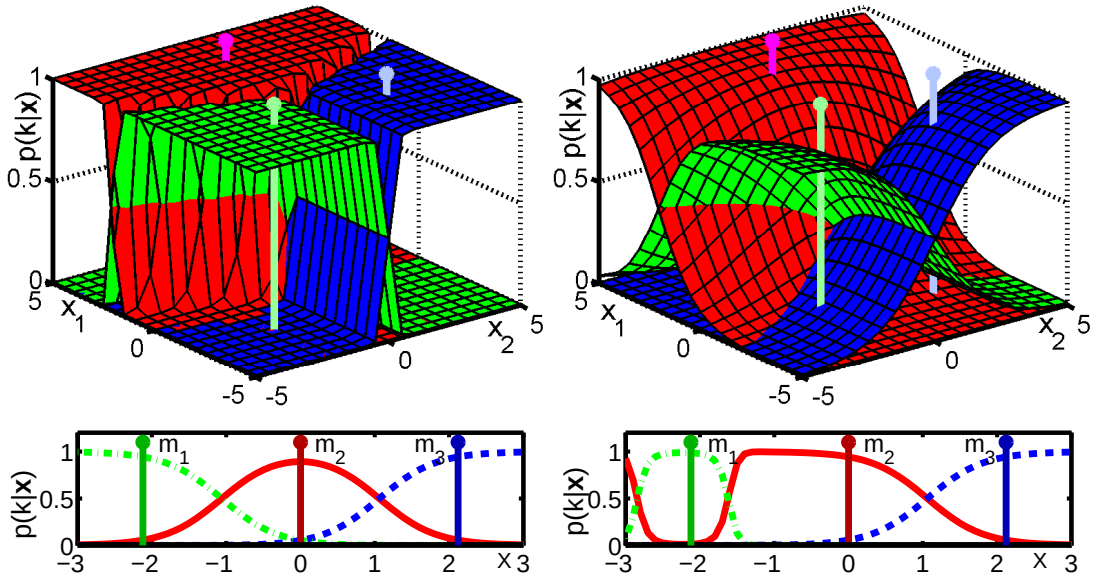


Figure 4.1: (*Top*) The membership probabilities given by equation (4.4) for three arbitrarily chosen $2D$ anchors and smoothing factor (*left*) $\sigma^2 = 1$ and (*right*) $\sigma^2 = 9$. (*Bottom*) The membership probabilities for $1D$ anchors given by (*left*) equation (4.4) with $\sigma^2 = 0.8$ and (*right*) equation (4.2) with $w_1 = w_2 = w_3$, $\sigma_1^2 = 0.04$, and $\sigma_2^2 = \sigma_3^2 = 0.8$. The anchors are marked with stems.

with equation (4.5) results in a cost function we seek to minimise with respect to σ:

$$\sigma = \arg\min_{\bar{\sigma}} \sum_{n=1}^{N} \left\| \boldsymbol{x}_n - \sum_{k=1}^{K} \frac{G(\boldsymbol{x}_n; \boldsymbol{m}_k, \bar{\sigma})}{\sum_{k'=1}^{K} G(\boldsymbol{x}_n; \boldsymbol{m}_{k'}, \bar{\sigma})} \boldsymbol{m}_k \right\|_2^2 \tag{4.6}$$

LLC methods minimise the quantisation loss w.r.t. the assignment coefficients $f_{\boldsymbol{m}}(\boldsymbol{x})$ in order to obtain a good linear combination of visual words $\boldsymbol{m}$ a.k.a. anchors that closely approximates descriptor $\boldsymbol{x}$. Additionally, various regularisation terms are enforced depending on a particular method. We realise that SA can also approximate descriptor $\boldsymbol{x}$ if a linear combination of anchors $\boldsymbol{m}_k$ weighted by the corresponding assignment coefficients $p(k|\boldsymbol{x})$ is performed over $k = 1, ..., K$. Therefore, we propose to employ such an approximation to evaluate the residual error of SA and to find σ that minimises it. Note that the membership probabilities $p(k|\boldsymbol{x})$ in figure 4.1 (top and bottom left) have almost linear slopes if σ is chosen appropriately. Moreover, these membership probabilities vary locally, *e.g.* varying descriptor x in figure 4.1 (bottom left) such that $-2 \leq x \leq 0$ induces only significant changes of the membership probabilities spanned by anchors $\boldsymbol{m}_1$ and $\boldsymbol{m}_2$. This makes SA somewhat similar to the LCC methods. However, if the GMM membership probabilities from equation (4.2) are used, the locality property becomes violated. This is illustrated in figure 4.1 (bottom right) by the red solid and green dashed curves. The slopes become ill-spanned and result in a poor approximation of descriptors in proximity of $\boldsymbol{m}_2$. The emphasis of the linear reconstruction is put on the descriptors in proximity of the narrow peak, despite these descriptors differing from each other only marginally. Therefore, the SA model for the membership probabilities from equation (4.2) may compromise the global reconstruction and prioritise it locally. The update rule for σ based on equation (4.3) is related to equation (4.6). However, the differences suggest that σ has two different meanings for: i) the optimal reconstruction of descriptor vectors measured by ξ^2, and ii) the density estimation problem.

Solving equation (4.6) is achieved by applying a coordinate-descent optimiser. Gradient and Hessian are computed on the cost function from equation (4.6):

$$\begin{aligned} \frac{\partial \xi^2}{\partial \sigma} &\approx [\xi^2(\sigma + \Delta\sigma) - \xi^2(\sigma - \Delta\sigma)]/2\Delta\sigma \\ \frac{\partial^2 \xi^2}{\partial \sigma^2} &\approx [\xi^2(\sigma + \Delta\sigma) + \xi^2(\sigma - \Delta\sigma) - 2\xi^2(\sigma)]/(\Delta\sigma)^2 \end{aligned} \tag{4.7}$$

Value of $\Delta\sigma$ depends on the descriptors used in the experiments outlined in the next section. It determines the quality of approximation of the gradient and is set arbitrarily to 1 and 0.001 for descriptors such that $\|\boldsymbol{x}\|_2 = 255$ and $\|\boldsymbol{x}\|_2 = 1$, respectively. Similarly to GMM, there is no closed form solution for equation (4.6). However, the cost function from this equation remains convex in σ. Moreover, only a small subset of descriptors from the training set requires evaluations to establish σ reliably. As every descriptor is represented by multiple visual words, a small subset of descriptors fills the entire vocabulary space with samples. This is illustrated in the experimental section.

4.4 Evaluations and Results

This section provides an experimental insight regarding the quality of the achieved descriptor approximations and the classification performance. Tests were performed on the PascalVOC10 Action Classification set [Everingham et al., 2010] (301 training, 307 validation, and 613 testing bounding boxes) and the Flower17 [Nilsback and Zisserman, 2008b] set (3 splits, each consisting of 680 training, 340 validation, and 340 testing images). For PascalVOC10, we report our results mainly on the validation set because its testing set is not publicly available. However, we also provide the test results of our approach submitted for the PascalVOC10 competition [Everingham et al., 2010]. Three descriptor variants were used to scrutinise the behaviour of the proposed cost function. The grey scale SIFT descriptors [Lowe, 1999] were extracted on PascalVOC10 with dense sampling on a regular grid. Intervals of 8, 14, 20, and 26 pixels, and patch radii of 16, 24, 32, and 40 pixels were applied. This produced 1200 descriptor vectors per image on average. For Flower17, the Opponent SIFT descriptors [van de Sande et al., 2008] combined with the Harris Laplace keypoints, as well as the segmentation-based descriptors from chapter 3 were extracted. These two descriptor variants resulted in 2300 vectors per image on average. The KDA and SVM classifiers were applied interchangeably to the χ^2 with RBF kernels (χ^2_{RBF}) [Tahir et al., 2009], as well as the linear kernels, both formed from SA histograms optimised according to the scheme proposed in section 4.3. The SPM approach [Lazebnik et al., 2006] with 3 levels of coarseness was also employed. The dictionaries with typical $K = 4000$ anchors were

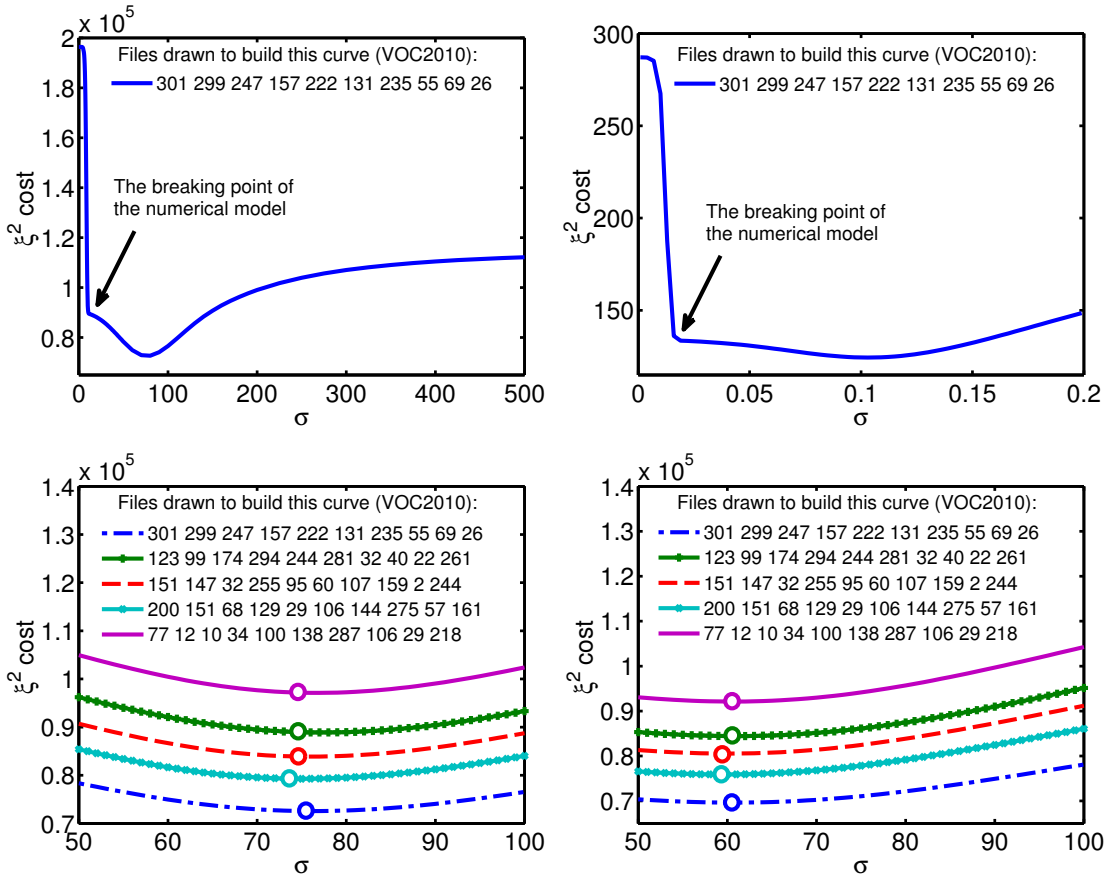


Figure 4.2: Experiments on the PascalVOC10 Action Classification set. (*Top*) The cost for a range of σ values given the grey SIFT descriptor such that (*left*) $\|\boldsymbol{x}\|_2 = 255$ (RSDS vocabulary) and (*right*) $\|\boldsymbol{x}\|_2 = 1$ (k-means vocabulary). (*Bottom*) The uncertainty of σ given $\|\boldsymbol{x}\|_2 = 1$ and (*left*) the RSDS and (*right*) the k-means vocabularies.

produced from the training sets by either Randomly Sampled Descriptor Set (RSDS), k-means, or solving GMM model according to equation (4.3).

First, we provide an empirical evaluation of the convexity of the ξ^2 cost from equation (4.6) with respect to σ. 10 training images were drawn at random from the PascalVOC10 set. Both RSDS and k-means were experimented with. In addition, the reconstruction error was evaluated as a function of the smoothing factor σ.

Figure 4.2 (top) illustrates the cost curves for the grey scale SIFT descriptors such that $\|\boldsymbol{x}\|_2 = 255$ (top left) and $\|\boldsymbol{x}\|_2 = 1$ (top right). The RSDS and k-means vocabularies were applied respectively. The produced curves show the quantisation error and illus-

trate several interesting properties of the proposed model: i) the numerical accuracy of the model becomes insufficient as σ moves to the left of the breaking point because the ratio of Gaussians in equation 4.4 becomes numerically unstable, ii) σ corresponding to the breaking point makes SA act closely to Hard Assignment, iii) there exists a unique minimum for the cost, and iv) as $\sigma \to \infty$, the assignment results in a total blurring: all descriptors are assigned to all K anchors with equal weights.

Figure 4.2 (bottom) illustrates how much the estimated σ varies with a subset of the drawn descriptors for RSDS (left) and k-means (right) vocabularies. Five-fold drawing process was employed, each time 10 unique images with the corresponding descriptor

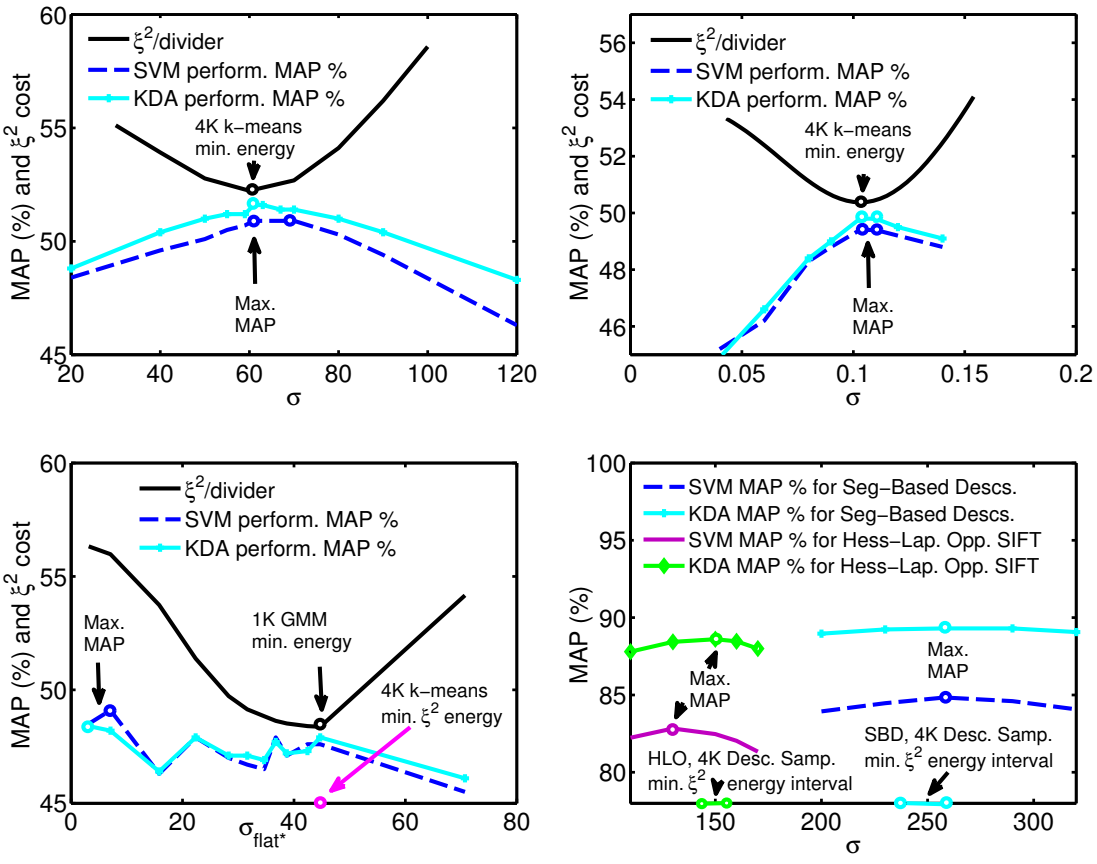


Figure 4.3: (*Top*) MAP maxima and ξ^2 minima (PascalVOC10, k-means, two variants of SIFT, SA from equation 4.4). (*Bottom left*) MAP maxima and ξ^2 for GMM given by equation (4.2). (*Bottom right*) MAP maxima and ξ^2 minima intervals on Flower17 (RSDS vocabulary, Opponent SIFT, the segmentation-based descriptors).

files were picked at random. Despite the absolute cost values differ, the minima are attained for approximately the same value of σ. The small uncertainty is negligible from the classification point of view. Note that the k-means vocabulary leads to the smaller quantisation loss compared to RSDS. Moreover, the optimal values of σ that linearise best the SA model differ for both types of visual dictionaries.

Figure 4.3 (top) presents the MAP performance and the quantisation error as functions of σ given the k-means vocabulary on the PascalVOC10 Action Classification dataset. KDA and SVM were applied to χ^2_{RBF} kernels. Both MAP and the energy ξ^2 were displayed in the same plot (ξ^2 is scaled to fit this plot). These both curves reveal a strong correlation between extrema of both measures. The optima are marked on curves with circles. The best classification performance was achieved for σ estimated according to equation (4.6). Plot 4.3 (top left) was prepared on the grey scale SIFT descriptor such that $\|\boldsymbol{x}\|_2 = 255$. Moreover, a scheme called Spatial Coordinate Coding was used that will be introduced in chapter 5.

Figure 4.3 (top right) was prepared with the grey scale SIFT such that $\|\boldsymbol{x}\|_2 = 1$. The RSDS dictionary and the SPM scheme with 3 levels of depth were used. The estimated σ proved to be optimal. Not shown in the plots, the RSDS vocabulary gave results about 0.5% MAP lower compared to k-means. Moreover, SA was additionally compared to SC. The same k-means dictionary was used as well as SPM. However, SC yielded only 48.7% whilst SA reached 49.4% MAP.

Figure 4.3 (bottom left) presents the MAP performance and ξ^2 for SA for equation (4.2). The full parameters estimated with GMM were used. The flattening σ^*_{flat} forces all $\sigma_k \leq \sigma^*_{flat}$ to $\sigma_k = \sigma^*_{flat}$. This parameter was varied to show the difference between non-uniform and uniform σ_k. When the majority of σ_k become equalised, ξ^2 drops. Gradually, the local MAP maxima align with the minimum of ξ^2.

Lastly, figure 4.3 (bottom right) presents MAP and the ξ^2 minima intervals on Flower17 set. The Opponent SIFT descriptor combined with the Harris Laplace detector, as well as the segmentation-based descriptor from section 3 were used. We combined KDA with χ^2_{RBF} and SVM with the linear kernels. The optimisation scheme proposed in equation (4.6) and SA from equation (4.4) were applied. For Opponent SIFT, the best

σ varied between 145 and 155 given different sets of 10 randomly drawn images used in estimations. The best MAP varied by up to 0.2% for σ estimated on less than 10 randomly picked images. σ estimated on the segmentation-based descriptors varied between 240 and 258 with 0.13% uncertainty in MAP.

The best results attained by us on the PascalVOC10 Action Classification dataset [Everingham et al., 2010] amount to 62.15% MAP and outperform other systems that are reported in [Everingham et al., 2010]. These results were attained by averaging multiple kernels computed on various descriptors [Tahir et al., 2009]. On Flower17, we obtained 89.3% MAP (85.4% accuracy) using the segmentation-based descriptor. For comparison, multiple kernel learning from [Yan et al., 2010] yields 86.7% accuracy.

4.5 Conclusions

We have presented a novel method for finding the optimal smoothing factor σ of the SA model. It is extensively demonstrated that the reconstruction error ξ^2 has a strong impact on the classification performance. Moreover, such a minimised quantisation loss correlates well with the best classification performance, as demonstrated on various descriptors and datasets. We have discussed relation of SA to GMM and the LCC methods. We conclude that finding the best performing smoothing factor helps linearise the SA model. Moreover, we demonstrated that the SA coder resulting from the simplified GMM model can challenge the standard GMM approach. The latter method requires in a large number of parameters which are harder to adjust and overcome overfitting. The proposed experiments led to the state-of-the-art results on both PascalVOC10 Action Classification and Flower17 datasets. In chapter 6, an improved assignment scheme benefiting from the foundations of this chapter will be proposed. Moreover, the SA model will be shown to further benefit from an appropriate pooling scheme.

Chapter 5

Spatial Coordinate Coding, Dominant Angle and Colour Pyramid Matching

Spatial Pyramid Matching lies at the heart of modern visual categorisation. Once the local image descriptors are transformed to vectors of visual words by the coding step, these features are further processed by the spatial pyramid with coarse-to-fine grids that quantise the spatial location of each descriptor associated with each feature. See section 1.2.2 for detailed illustration of SPM. However, such a representation results in extremely large histogram vectors of $200K$ or more elements, increasing both computational and memory requirements. This chapter investigates alternative ways of introducing the spatial information during formation of the histogram representations. Specifically, we propose to apply spatial coordinates of descriptor keypoints at the descriptor level. We refer to such an approach as Spatial Coordinate Coding. Furthermore, vertical or horizontal information, radius, or angle is used to perform semi-coding. This is achieved by adding one of the spatial components at the descriptor level whilst applying SPM to the another component. Moreover, we demonstrate that the Pyramid Matching scheme can be applied robustly to other measurements: so-called Dominant Angle and colour. We demonstrate state-of-the art results with means of the popular coding techniques such as Soft Assignment, explained in chapter 4, and Sparse Coding.

5.1 Introduction

Spatial Pyramid Matching (SPM) proposed in [Lazebnik et al., 2006] has been employed by the majority of the modern visual categorisation systems. SPM is an extension of Pyramid Match Kernel (PMK) proposed in [Grauman and Darrell, 2005]. SPM instantly became a popular method to incorporate the spatial information into the classification process. Popular systems that apply SPM are: Soft Assignment with the χ^2 distance combined with RBF kernel (χ^2_{RBF}) [van Gemert et al., 2010, Tahir et al., 2009], Linear Coordinate Coding from [Yu et al., 2009], Sparse Coding from [Yang et al., 2009], Locality-constrained Linear Coding from [Wang et al., 2010], and approaches using Fisher Vector Encoding [Perronnin et al., 2010] or Super Vector Coding [Zhou et al., 2010]. Note that the last two approaches produce extremely large histograms, which are further extended with the SPM scheme to improve their performance. Such approaches use a simplified layout of spatial partitions, *e.g.* the method from [Tahir et al., 2009] used 1×1, 2×2, 1×3 horizontal, and 3×1 vertical windows whilst [Marszalek et al., 2007, Zhou et al., 2010] used $1\times1, 2\times2$, and 1×3 horizontal divisions.

For the first contribution, we propose a scheme called Spatial Coordinate Coding (SCC) that applies spatial coordinates from the descriptor keypoints at the descriptor level. This reduces the histogram sizes from $K\times S(Q^2)$ to $K\times S(1^1)$, where K is the size of input histograms, Q is the number of SPM levels of quantisation, and $S(Q^l) = \sum_{q=1}^{Q} q^l$. Moreover, we manipulate spatial coordinates to be partially absorbed at the descriptor level and by SPM, and reduce the histogram sizes from $K\times S(Q^2)$ to $K\times S(Q^1)$. SCC is demonstrated to work with two popular descriptor coding methods: SA and SC. It can be also applied to methods proposed in [Perronnin et al., 2010, Zhou et al., 2010].

For the second contribution, we apply Pyramid Matching to various types of measurements. The Dominant Angle (DA) mechanism proposed in [Lowe, 1999] can be applied instead of the spatial information by DA normalising the local image descriptors and applying partitioning based on the values of DA rather than the spatial coordinates. The colour information of the segmentation-based descriptors from chapter 3 is exploited in a similar manner. Furthermore, we demonstrate that SCC, DoPM, and CoPM deliver the state-of-the-art results on two datasets.

5.2 Spatial Coordinate Coding

SA and SC are two extremely popular techniques for transforming the descriptors into the image signatures in the BoW model. For convenience, the notion of the image signatures is explained in section 1.2.2. Typically, systems that employ these methods add the spatial information to the classification process by SPM. However, this results in the signatures of length $K \times S(Q^2)$.

Let $\boldsymbol{x}^s = [c^x/w, c^y/h]^T$ be the spatial coordinates of descriptor $\boldsymbol{x}$ that are normalised by the image width and height. Furthermore, let $\boldsymbol{x}^p = [r, \phi]^T$ be a vector with the unit normalised radius $r = \sqrt{(c^x/w - 0.5)^2 + (c^y/h - 0.5)^2}/(\sqrt{2}/2)$ as well as angle information $\phi = 0.5 + \text{atan}(c^x/w - 0.5,\, c^y/h - 0.5)/(2\pi)$. Let $\boldsymbol{m}_k$ be the visual words of a visual vocabulary $\mathcal{M}$ with K atoms, such that $k = 1, ..., K$. Moreover, let $\mathcal{M}$ to be built by either k-means or Randomly Sampled Descriptor Set (RSDS). Let $\boldsymbol{m}_k^s$ and $\boldsymbol{m}_k^p$ be the corresponding elements of the spatial vocabulary arranged in the same manner as parametrisations $\boldsymbol{x}^s$ and $\boldsymbol{x}^p$, respectively.

We propose to replace the SPM scheme in the BoW model by applying either: i) spatial parametrisation $\boldsymbol{x}^{'} = \boldsymbol{x}^s/2$ or $\boldsymbol{x}^{'} = \boldsymbol{x}^p/2$ leading to the signatures of length $K \times S(1^1)$, or ii) semi-spatial parametrisation $x^{'} = c^x/w$, $x^{'} = c^y/h$, $x^{'} = r$, or $x^{'} = \phi$. In the latter case, both spatial channels, *e.g.* c^x/w and c^y/h are processed one by the SCC and the other by the SPM scheme. The same arrangement is used for r and θ. This leads to the signatures of length $K \times S(Q^1)$, which are shorter compared to the standard SPM scheme resulting in the signatures of length $K \times S(Q^2)$, as indicated earlier.

5.2.1 SCC for Soft Assignment

The Soft Assignment model is introduced in chapter 4. Enhancing formula (4.4) with the spatial or semi-spatial coordinates can be done by adding spatially parametrised vectors $\boldsymbol{x}^{'}$ and $\boldsymbol{m}_k^{'}$ to the Gaussian components as follows:

$$G^{'}(\boldsymbol{x}, \boldsymbol{x}^{'}; \boldsymbol{m}, \boldsymbol{m}^{'}, \sigma^{'}, \omega) = G\left((1-\omega)\boldsymbol{x}; (1-\omega)\boldsymbol{m}, \sigma^{'}\right) \cdot G(\omega \boldsymbol{x}^{'}; \omega \boldsymbol{m}^{'}, \sigma^{'}) \tag{5.1}$$

The additional parameter $\omega \in \langle 0, 1)$ represents a trade-off between the visual appearances and the spatial bias. Redefining the membership probability from equation (4.4)

results in:

$$p(k|\boldsymbol{x},\boldsymbol{x}^{'}) = \frac{G^{'}(\boldsymbol{x},\boldsymbol{x}^{'};\boldsymbol{m}_k,\boldsymbol{m}_k^{'},\sigma^{'},\omega)}{\sum_{k'=1}^{K} G^{'}(\boldsymbol{x},\boldsymbol{x}^{'};\boldsymbol{m}_{k'},\boldsymbol{m}_{k'}^{'},\sigma^{'},\omega)} \tag{5.2}$$

The best smoothing factor $\sigma^{'}$ differs from σ for such a reformulated model in equation (5.2) due to the additional spatial information being introduced. One can use cross-validation to find the best $\sigma^{'}$ or employ the optimisation method from chapter 4.

5.2.2 SCC for Sparse Coding

The operating principle of SC is to express each descriptor vector as a sparse linear combination of visual words. The ℓ_1 norm computed over the assignments favours only a small subset of non-zero coefficients during the assignment step. This is known as the Lasso problem. The BoW model employing such an assignment step and SPM was shown to perform well in [Yang et al., 2009]. Finding the sparse assignments for descriptor $\boldsymbol{x}$ given visual vocabulary $\mathcal{M} \in \mathbb{R}^{D \times K}$ is achieved by optimising the following:

$$\boldsymbol{\phi} = \arg\min_{\bar{\boldsymbol{\phi}}} \left\| \boldsymbol{x} - \mathcal{M}\bar{\boldsymbol{\phi}} \right\|_2^2 + \alpha \|\bar{\boldsymbol{\phi}}\|_1 \tag{5.3}$$

Parameter α regulates the sparsity of the assignment vector $\boldsymbol{\phi}$. We propose to enhance formula (5.3) with the spatial or semi-spatial coordinates by introducing vectors $\boldsymbol{x}^{'}$ and $\boldsymbol{m}_k^{'}$ to the Lasso problem. This is achieved by adding a second quantisation loss that controls the quantisation cost of the spatial components:

$$\boldsymbol{\phi} = \arg\min_{\bar{\boldsymbol{\phi}}} (1-\omega) \left\| \boldsymbol{x} - \mathcal{M}\bar{\boldsymbol{\phi}} \right\|_2^2 + \omega \left\| \boldsymbol{x}^{'} - \mathcal{M}^{'}\bar{\boldsymbol{\phi}} \right\|_2^2 + \alpha \|\bar{\boldsymbol{\phi}}\|_1 \tag{5.4}$$

Note that equations (5.2) and (5.4) can be solved with standard SA and SC in equations (4.4) and (5.3), respectively, by simply concatenating appropriately the local image descriptors $\boldsymbol{x}$ with the corresponding spatial coordinates $\boldsymbol{x}^{'}$. Specifically, we perform an augmentation of descriptor $\boldsymbol{x}$ such that $\boldsymbol{x} := \left[\sqrt{1-\omega}\boldsymbol{x}^T, \sqrt{\omega}(\boldsymbol{x}^{'})^T\right]^T$. The analogous operation has to be performed on visual words $\boldsymbol{m}_k$ for all $k = 1, ..., K$.

5.3 Dominant Angle and Colour Pyramid Matching

This section provides details on how to exploit Dominant Angle (DA) from [Lowe, 1999], as well as the colour channels from the segmentation-based descriptors introduced in

Figure 5.1: Illustration of the spatial bias in images.

chapter 3, in the Pyramid Matching scenario. A variety of cues may be appropriate for quantising them at multiple levels of coarseness. The spatial bias introduced in chapter 1 is illustrated for convenience in figure 5.1. Note that various image partitions tend to contain different visual appearances. For instance, the sun and clouds usually appear in the sky. Therefore, they are mostly to appear in the upper parts of images.

If spatial locations of objects of class $s \in \mathcal{S}$ introduce any spatial bias, this can be captured in a set of spatial coordinates $\mathcal{X}_s^{'}$ associated with class s. Subsequently, observing object o at location $\boldsymbol{x}^{'} \in \mathcal{X}_s^{'}$ in a previously unseen image increases belief that this object belongs to class s. If $p(o{=}s)$ represents a belief of a given recognition system that object o belongs to class s, then spatial location $\boldsymbol{x}^{'}$ of this object can alter such a belief, *e.g.* $p(o{=}s|\boldsymbol{x}^{'} \in \mathcal{X}_s^{'}) \geq p(o{=}s) \geq p(o{=}s|\boldsymbol{x}^{'} \notin \mathcal{X}_s^{'})$.

Similar can be said about the orientations of dominant edges in images. For instance, trunks of trees and fences are more likely to maintain vertical positions in image collections. This phenomenon is illustrated in figure 5.2 and called as *the orientation bias*. It can be captured by the DA mechanism build into the SIFT descriptors. DA is a direction with respect to the origin of an image patch that indicates the orientation of the largest image gradients within that patch. Capturing a set of DA called Θ_s

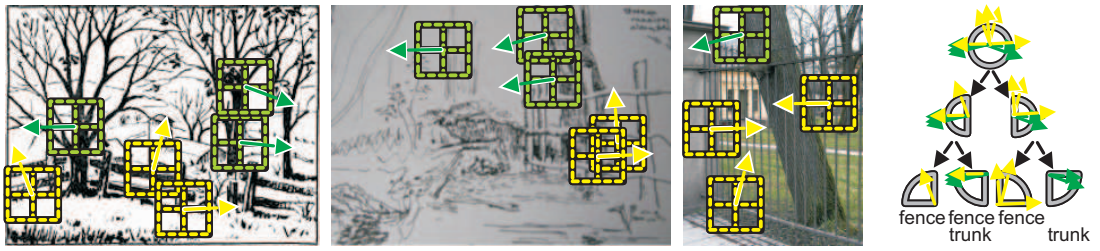


Figure 5.2: Illustration of the orientation bias in images.

which is associated with class s should results in an analogous sequence of believes, as illustrated earlier for the spatial bias: $p(o=s|\theta \in \Theta_s) \geq p(o=s) \geq p(o=s|\theta \notin \Theta_s)$. Note that the rotationally variant descriptors result in much better classification results on the segmentation-based descriptor in chapter 3 compared to rotationally invariant counterparts. The similar observation holds for the SIFT descriptors. This suggests that the orientation bias in images is essential in robust visual categorisation.

Moreover, the facial complexion or a fur of animals are likely to be of a limited colour range. Figure 5.3 illustrates that the sky and trees can be partially distinguished from each other by their colour appearances. Therefore, one can capture *the colour bias* in images by building a set of colours associated with class s, called $\mathcal{C}_s$. This should lead to the sequence of believes such that: $p(o=s|\boldsymbol{c}\in\mathcal{C}_s) \geq p(o=s) \geq p(o=s|\boldsymbol{c}\notin\mathcal{C}_s)$.

In the following experiments, we introduce DA to the classification process in two ways: i) by setting $x^{'} = \theta$, or ii) by performing Pyramid Match on θ. Regarding colour, the segmentation-based descriptors are used as they contain the colour statistics. We reduced $20D$ opponent colour histograms by Principal Component Analysis (PCA) to $10D$. An opponent colour component corresponding to the highest variance after the projection was processed with Pyramid Matching. The remaining 9 components replaced the original opponent vectors.

5.4 Evaluations and Results

This section provides an evaluations of Spatial Coordinate Coding and Spatial Pyramid Matching. The evaluations were carried out on the PascalVOC10 Action Classification set from [Everingham et al., 2010] (301 training, 307 validation, and 613 testing bounding boxes) and the Flower17 set from [Nilsback and Zisserman, 2008b] (3 splits

Figure 5.3: Illustration of the colour bias in images.

of data, each consisting of 680 training, 340 validation, and 340 testing images). For PascalVOC10, we report the results obtained on the validation set, as the testing set is not publicly available. Moreover, we quote our results on the test set submitted for the PascalVOC10 competition [Everingham et al., 2010]. Experiments on Dominant Angle Pyramid Matching were performed on the PascalVOC07 set [Everingham et al., 2010].

Two variants of descriptors were exploited: the grey scale SIFT for the PascalVOC10 and PascalVOC07 sets, as well as the segmentation-based descriptors introduced in chapter 3 for the Flower17 set. Dense sampling on a regular grid with the intervals of 8, 14, 20, and 26 pixels, and patch radii of 16, 24, 32, and 40 pixels was applied for SIFT. This produced approximately 1200, 3690, and 2300 descriptors per image given the PascalVOC10, PascalVOC07, and Flower17 sets, respectively. We combined the KDA classifier [Tahir et al., 2009] with the χ^2 distance used by RBF kernels (χ^2_{RBF}) and SVM with the linear kernels. These kernels were formed with either SA introduced in chapter 4 or SC [Yang et al., 2009].

As a reference, SPM with 3 and 4 levels of depth were employed for SA and SC respectively. The visual vocabulary of size $K = 4000$ was produced by k-means for the PascalVOC10 and pascalVOC07 sets, whilst the RSDS vocabulary was extracted for Flower17. RSDS performed better than k-means on this set. Nonetheless, this chapter is not concerned with investigations of various kinds of visual dictionaries.

5.4.1 SCC and Action Classification

The Pascal 2010 Action Classification set is provided with the bounding boxes that delineate humans performing various actions. Every person's head is roughly aligned to the top middle location of a given bounding box. Therefore, the spatial locations of objects interacted with can be expressed with respect to the top middle reference point.

SC_{1234}	SC+SCC	SA_{123}	SA+SCC	SA+SCC
Lin+SVM	Lin+SVM	χ^2_{RBF}2+KDA	χ^2_{RBF}+KDA	χ^2_{RBF}+KDA
1ker+val	1ker+val	1ker+val	1ker+val	multiker+tst
50.6	49.0	49.8	**51.6**	**62.15**

Table 5.1: MAP for the PascalVOC10 Action Classification set.

To exploit this, Spatial Coordinate Coding is applied and compared with Spatial Pyramid Matching. Table 5.1 presents the results obtained on this set. SC combined with SPM given 4 levels of coarseness (denoted SC_{1234}) turned out to be a better performer than SA and SPM with 3 levels of coarseness (denoted as SA_{123}). SC combined with the SCC scheme (denoted as $SC{+}SCC$) performed marginally worse than SC_{1234}. SA with SCC (denoted as $SA{+}SCC$) was the strongest performer reaching 51.6% MAP. Moreover, a combination of multiple kernels as in [Tahir et al., 2009] led to the state-of-the-art score of 62.15% MAP on the testing set. We observed that SA with the χ^2_{RBF} kernel benefits the most from the proposed SCC scheme.

5.4.2 Understanding the Dominant Angle

The PascalVOC07 set consists of 20 object categories that result in high variability in scale, rotation, and spatial positions. This section presents a brief study on the Dominant Angle and its applicability in Pyramid Matching. The results reported below were achieved with SA, the χ^2_{RBF} kernel, and the KDA classifier. According to table 5.2, DA is an important modality aiding robust visual categorisation. DA Inv. denotes the baseline obtained with the SIFT descriptors that were deemed invariant to the patch rotation. Enabling such an invariance decreased the classification performance compared to the typical DA variant scenario (DA Var.) applied in the BoW model from 50.23% to 46% MAP. Moreover, we used DA invariant SIFT and injected DA directly to equation 5.2 (referred to as DACC) with $\omega = \frac{1}{2}$, $\frac{2}{3}$, and $\frac{4}{5}$. DACC with $\omega = \frac{4}{5}$ achieved 50.24% MAP on a par with the DA Var. scenario. Therefore, DA is shown as a very important cue as, by removing DA information from SIFT and then reintroducing DA back to the classification process, it regained its full performance. For comparison, DA Var. and DACC with $\omega = \frac{4}{5}$ are indicated in green in table 5.2.

DA Inv.	DA Var.	DACC$\omega{=}\frac{1}{2}$	DACC$\omega{=}\frac{2}{3}$	DACC$\omega{=}\frac{4}{5}$
46.00	**50.23**	47.2	49.80	**50.24**
DA Var.+ SPM 54.3	DA_{12468} 52.30	DA_{136912} **53.40**	DA_{136912}+ SPM **56.3**	

Table 5.2: MAP for the PascalVOC07 set illustrating relevance of DA.

SA Lin SVM	SCC$\omega = \frac{6}{11}$ 84.31	SCC$\omega = \frac{9}{14}$ 84.96	SPM 86.8	SPM$r\theta$ 85.6
SA χ^2_{RBF} KDA	SCC$\omega = \frac{6}{11}$ 90.96	SCC$\omega = \frac{9}{14}$ **91.16**	SPM 89.3	SPM$r\theta$ 89.63
SC Lin SVM	NO SCC 89.11	SCC$\omega = \frac{1}{3}$ **90.46**	SPM **90.83**	

Table 5.3: MAP for the Flower17 set comparing the SCC and SPM schemes.

Furthermore, DA can be quantised with Pyramid Matching. The orientation invariant SIFT descriptor combined with Pyramid Matching with 5 levels of angular splits 1, 3, 6, 9, 12 (denoted as DA_{136912}) achieved 53.4% MAP. This constitutes a 3.1% improvement over the DA Var. scenario. We attribute this to exploiting the orientation bias at the multiple levels of coarseness. Moreover, combining DoPM with SPM (denoted as DA_{136912}+SPM) boosted performance from 54.3% to 56.3% MAP given the grey scale SIFT descriptor only. The best results for DoPM are indicated in red in table 5.2.

5.4.3 SsCC and CoPM on Flower17

Performance of both SCC and Semi-spatial Coordinate Coding (SsCC) was evaluated on the Flower17 set with means of both SA and SC. According to results in table 5.3, SA with SVM and the linear kernel (SA Lin SVM row) achieved better results of 86.8% MAP if using SPM with 3 levels of depth rather than SCC. Radius and θ parametrised SPM (SPM$r\theta$) was a close performer. Also, SCC$\omega = \frac{9}{14}$ achieved a similar score of 84.93% MAP. The gap of 1.9% in performance between these two methods is bridged by Semi-spatial Coordinate Coding presented in table 5.4. Note that SA with the χ^2_{RBF} kernel and KDA classifier (SA χ^2_{RBF} KDA row) exploited SCC to its fullest

SA Lin SVM	SPMy+ SCCx **87.5**	SPMx+ SCCy **87.1**	SPMθ+ SCCr **87.5**	SPMr+ SCCθ **87.5**
SA χ^2_{RBF} KDA	SPMy+ SCCx 90.4	SPMx+ SCCy 90.1	SPMθ+ SCCr 90.2	SPMr+ SCCθ 90.2
SC Lin SVM	SPMy+ SCCx 90.7	SPMx+ SCCy **91.3**	SPMθ+ SCCr **91.2**	SPMr+ SCCθ 90.4

Table 5.4: MAP for the Flower17 set utilising Semi-spatial Coordinate Coding.

potential outperforming SPM by approximately 1.8% and reducing the histogram sizes from $4K \times S(3^2) = 56K$ to $4K \times S(1^1) = 4K$ elements. Moreover, SC with the linear kernel and SVM (SC Lin SVM row) benefited 1.2% from SCC over the no-spatial-information scenario (NO SCC), whilst SPM led to about 1.6% over NO SCC. These results are further improved by the SsCC scheme, as presented below.

According to table 5.4 (first row), all semi-spatial combinations improved results by up to 0.7% over SA with the linear kernel and SPM. We combined SPM and SCC for specific semi-spatial channels x, y, r, θ, as proposed in section 5.2. SA with the χ^2_{RBF} kernel and the KDA classifier (second row) favours full SCC reaching 91.16% compared to 90.4% MAP for SPMy+SCCx. SC (bottom row) benefited from the semi-spatial variants SPMx+SCCy and SPMθ+SCCr scoring 91.3% MAP and outperforming SPM by 0.47%. This limited the signature sizes from $4K \times S(4^2) = 120K$ to $4K \times S(3^1) = 24K$.

Moreover, we investigated the benefit of quantising the colour cues on Flower17. The setup for this experiment is explained in section 5.3. SA with the χ^2_{RBF} kernel, KDA classifier, and SCC (91.16% MAP, 86.4% accuracy) were enhanced by this method and produced the state-of-the-art results of 92.2% MAP (87.4% accuracy). This, combined with the Opponent SIFT descriptor at the kernel level, increased results to 95.2% MAP (91.4% accuracy). The runner-up reports 86.7% accuracy [Yan et al., 2010].

5.5 Conclusions

We have presented a novel method injecting the spatial coordinates to the classification process at the descriptor level. This resulted in small image representations and improved results for Soft Assignment combined with the χ^2_{RBF} kernels. Moreover, a semi-spatial approach was proposed to benefit Sparse Coding combined with the linear kernels. Previously overlooked importance of the Dominant Angle mechanism was demonstrated together with the promising classification results. This was especially prominent when DA was combined with Pyramid Matching. As various objects exhibit different levels of the colour constancy, we showed that the opponent colour components also thrive on quantising with Pyramid Matching. This resulted in the state-of-the-art performance on both PascalVOC10 Action Classification and Flower17 sets.

Chapter 6

Comparison of Mid-Level Feature Coding Approaches And Pooling Strategies in Visual Concept Detection

Bag-of-Words lies at the heart of modern object category recognition systems. After descriptors are extracted from images, they are expressed as vectors representing visual word content, referred to as mid-level features. In this chapter, we review a number of techniques for generating mid-level features, including Soft Assignment, Locality-constrained Linear Coding, Sparse Coding, and propose Approximate Locality-constrained Soft Assignment. Next, we also identify the underlying properties that affect their performance. Moreover, we investigate various pooling methods that aggregate mid-level features into vectors representing images. Average, Max-pooling, and a family of likelihood inspired pooling strategies are scrutinised. We demonstrate how both coding schemes and pooling methods interact with each other. We generalise the investigated pooling methods to account for the descriptor interdependence and introduce an intuitive concept of improved pooling. We also propose a coding-related improvement to increase its speed. Lastly, state-of-the-art performance in classification

is demonstrated on Caltech101, Flower17, ImageCLEF11, and PascalVOC07 datasets.

6.1 Introduction

Bag-of-Words proposed in [Sivic and Zisserman, 2003, Csurka et al., 2004] is a popular approach which transforms local image descriptors [Lowe, 1999, Mikolajczyk and Schmid, 2005, van de Sande et al., 2008] into image representations that are used in matching and classification. Its first implementations were associated with object retrieval and scene matching [Sivic and Zisserman, 2003], as well as visual categorisation [Csurka et al., 2004]. The BoW approach has undergone significant changes over recent years but it can be summarised by the following steps:

1) First, the local image descriptors are extracted from images. Next, a dictionary, also known as a visual vocabulary, is learnt by finding a set of descriptive discrete appearance prototypes defined in the descriptor space, *e.g.* by k-means clustering of descriptors from a training dataset. These prototypes are often called as visual words, centres, atoms, and anchors.

2) Feature coding a.k.a. mid-level coding is then performed by embedding local descriptors into the visual vocabulary space. This results in so-called mid-level features which express each descriptor by a subset of visual words.

3) A pooling step is carried out to transform mid-level features from an image into a final representation in a form of vector called image signature. A basic pooling approach aggregates every local descriptor represented by a combination of visual words into a single signature vector. Finally, training and classification can be performed on the signatures by a classifier, *e.g.* SVM [Cortes and Vapnik, 1995] or KDA [Tahir et al., 2009].

Each step has a strong impact on the quality of image representation and can affect the classification performance and computational speed. The objective of this chapter is to closely examine various techniques proposed for the coding and pooling steps and demonstrate their performance in a number of benchmarks.

A baseline BoW approach [Sivic and Zisserman, 2003] employs k-means clustering of local descriptors from a training dataset and assigning each descriptor to the nearest cluster (mid-level coding). This is often referred to as Hard Quantisation or Hard Assignment. A histogram representing the image is obtained by counting the number of assignments per cluster. Averaging such counts by the number of descriptors in the image results in Average pooling [Csurka et al., 2004, van Gemert et al., 2008, 2010].

A number of mid-level coding methods proposed to date include Kernel Codebook [van Gemert et al., 2008, 2010, Philbin et al., 2008, Lingqiao et al., 2011] a.k.a. Soft Assignment and Visual Word Uncertainty, the family of Linear Coordinate Coding, entailing Sparse Coding (*e.g.* Lasso [Lee et al., 2007, Yang et al., 2009] and greedy coders like Match Pursuit [Mallat and Zhang, 1993] and Orthogonal Match Pursuit [Tropp, 2004]), Linear Coordinate Coding [Yu et al., 2009], Locality-constrained Linear Coding [Wang et al., 2010], Laplacian Sparse Coding [Gao et al., 2010], and Over-Complete Sparse Coding [Yang et al., 2010]. Other robust approaches include Fisher Vector Encoding [Perronnin and Dance, 2007, Perronnin et al., 2010], Super Vector Coding [Zhou et al., 2010], Vector of Locally Aggregated Descriptors [Jégou et al., 2010], and Vector of Locally Aggregated Tensors [Negrel et al., 2012].

Quantisation effects in Hard Assignment coding were found to be a source of ambiguity [Philbin et al., 2008]; descriptor vectors lying on the border of two clusters can be assigned to one or the other merely due to low-level stochastic noise. It is argued in [Wang, 2007] that a small set of descriptors along cluster boundaries are the most discriminative ones and must be represented well, *e.g.* by hierarchical clustering. The quantisation effect is somewhat alleviated by assigning descriptors to their l-nearest clusters [Philbin et al., 2008, Tahir et al., 2009] rather than to the nearest cluster only. However, descriptor vectors can be different and yet they may share the same l-nearest clusters. Soft Assignment is another approach to feature coding [van Gemert et al., 2008, 2010] that yields cluster membership probabilities for every visual word given a descriptor. Such a strategy is beneficial as descriptors are assigned to every cluster centre with different probabilities thus improving the quantisation properties of the coding step. Lastly, there has been a significant progress in Linear Coordinate Coding methods [Lee et al., 2007, Yang et al., 2009, Yu et al., 2009, Wang et al., 2010, Gao

et al., 2010, Zhou et al., 2010] leading to state-of-the-art results with BoW [Everingham et al., 2010]. These approaches seek a few weighting coefficients to linearly combine elements of the dictionary to approximate a given descriptor. Final image signatures are formed from the largest coefficients per visual word which is termed Max-pooling [Yang et al., 2009, Boureau et al., 2010a,b, Lingqiao et al., 2011].

Recent progress in mid-level feature coding has also provided an insight into the role played by pooling during the generation of image signatures. The theoretical relation between Average and Max-pooling was studied in [Boureau et al., 2010a]. A detailed likelihood-based analysis of feature pooling was conducted in [Boureau et al., 2010b] which led to a *theoretical expectation of Max-pooling*, improving overall classification results. Power Normalisation has been also applied to Average pooling by Fisher Vector Encoding [Perronnin et al., 2010]. Lastly, Max-pooling has been recognised as a lower bound on the likelihood of *at least one particular visual word being present in an image* [Lingqiao et al., 2011]. We show later that some of these methods are closely related.

A crucial component of the BoW approach, which has an impact on pooling, is Spatial Pyramid Matching [Lazebnik et al., 2006]. It exploits spatial bias in images by expressing spatial relations at multiple levels of quantisation. Also, clustering mid-level features and applying pooling in each cluster [Boureau et al., 2011] limits the uncertainty of pooling. Exploiting other types of bias in images to partition the features is also effective, *e.g.* Dominant Angle and Colour Pyramid Matching from chapter 5.

A recent review of coding schemes [Chatfield et al., 2011] includes Hard Assignment, Soft Assignment, Approximate Locality-constrained Linear Coding, Super Vector Coding, and Fisher Vector Encoding. Evaluations of BoW in [Yang et al., 2007] employ ideas from text analysis: term frequency, inverse document frequency and various normalisation schemes. The importance of mid-level coding versus dictionary training is studied in [Coates and Ng, 2011]. Various dictionary learning approaches are considered and described in [Tosic and Frossard, 2011]. Lastly, Hard Assignment, Soft Quantisation, and Sparse Coding are combined with Average and Max-pooling, and their characteristics are studied in depth in [Boureau et al., 2010a]. More pooling strategies are presented in [Boureau et al., 2010b].

Although there exist various comparisons of BoW, there is a lack of large scale evaluation of both mid-level coding and pooling strategies in a common testbed. The analysis of interaction between these two stages constitutes the main contribution of our work:

1) We evaluate various mid-level coding schemes such as Soft Assignment (SA) [van Gemert et al., 2008, 2010, Philbin et al., 2008], its extension Approximate Locality-constrained Soft Assignment (LcSA) proposed by [Lingqiao et al., 2011] as well as by us[1], Sparse Coding (SC) [Lee et al., 2007, Yang et al., 2009], and Approximate Locality-constrained Linear Coding (LLC) from [Wang et al., 2010].

2) We compare various pooling schemes such as Average pooling (Avg) used in [Csurka et al., 2004, van Gemert et al., 2008, 2010], Max-pooling (Max) used in [Yang et al., 2009, Boureau et al., 2010a,b, Lingqiao et al., 2011] , Power Normalisation a.k.a. Gamma Correction (Gamma) used in [Perronnin et al., 2010], *theoretical expectation of Max-pooling* (MaxExp) proposed in [Boureau et al., 2010b], the probability of *at least one particular visual word being present in an image* (ExaPro) proposed in [Lingqiao et al., 2011], ℓ_p norm (lp-norm) as a trade-off between Average and Max-pooling explored in [Boureau et al., 2010b], and Mix-order Max-pooling (MixOrd) from [Lingqiao et al., 2011].

3) We devise a simple approximation of MaxExp pooling (AxMin) and illustrate that Gamma also approximates MaxExp. Before evaluating MaxExp, AxMin, as well as Gamma, we generalise them to account for the descriptor interdependence, *e.g.* due to the overlap of descriptors. A pooling extension is proposed that uses the top n largest mid-level feature coefficients (@n) per visual word. This reduces the noise and improves the performance. We show that Max-pooling is a special case of @n.

4) Spatial (SPM) and Dominant Angle Pyramid Matching (DoPM), introduced in [Lazebnik et al., 2006] and chapter 5 respectively, are employed to demonstrate their interaction with the pooling step. The early fusion of the spatial cues and descriptors called Spatial Coordinate Coding (SCC) from chapter 5 is used, as it leads to 36× faster kernel computations compared to SPM.

[1] This contribution was independently proposed and developed shortly before a similar approach was published by others in [Lingqiao et al., 2011].

5) Finally, the role of the reconstruction error a.k.a. quantisation error in the coding schemes is illustrated. It is demonstrated empirically that minimising such an error over parameters of LcSA correlates well with its best classification performance. To increase the efficiency of coding, two coding methods are combined with Spill Trees [Liu et al., 2004] and compared to the baseline methods of various dictionary sizes.

Section 6.2 formally introduces Bag-of-Words and describes mid-level coding methods. Section 6.3 introduces pooling methods. Section 6.4 details the experimental framework. Various coding and pooling methods are then compared, followed by a detailed discussion. Section 6.5 draws conclusions on this work.

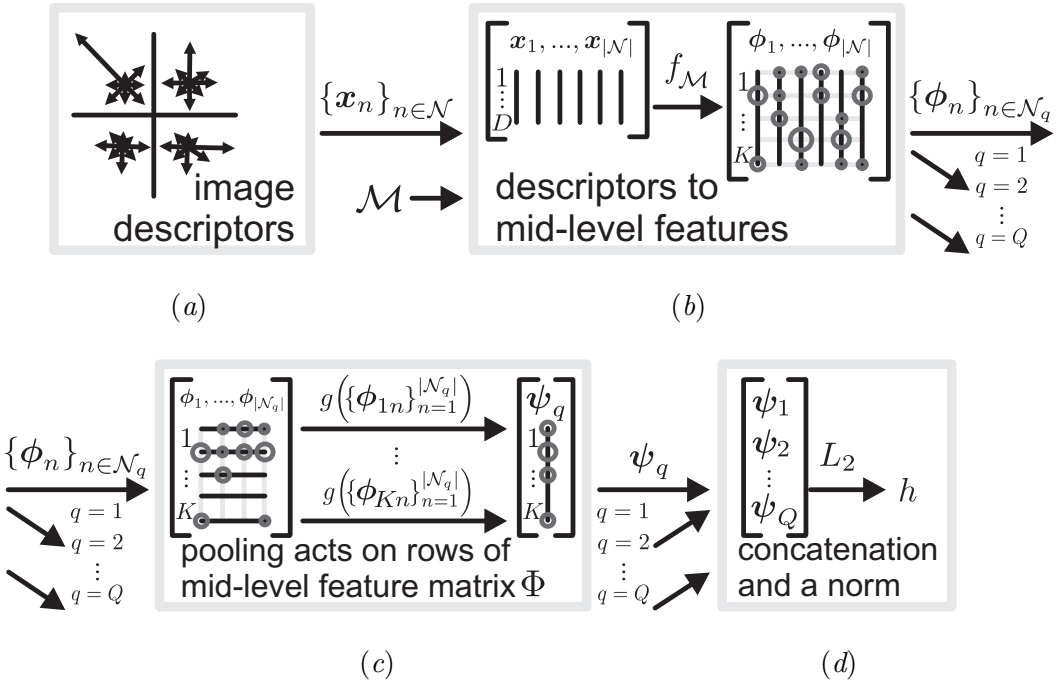


Figure 6.1: Overview of Bag-of-Words showing mid-level coding and pooling steps. (a) $|\mathcal{N}|$ local descriptors of dimension D are extracted from an image. (b) Mid-level coding embeds the descriptors into the visual vocabulary space using K visual words from dictionary $\mathcal{M}$. Circles of various sizes illustrate values of mid-level coefficients. (c) Mid-level features of partition q are stacked. Next, pooling aggregates the values along rows and forms a single vector per spatial partition. (d) Vectors from all partitions are concatenated and normalised to form signature h.

6.2 Overview of Mid-level Feature Coding Approaches

The goal of mid-level coding is to embed descriptors in a representative visual vocabulary space. This can be seen as a form of interpolation. Mid-level coding interpolates data on an irregular grid stretched across the surface of a hypersphere of ℓ_2 norm normalised descriptor space. Due to the high dimensionality of the descriptor space, it is not practical to partition it evenly [Tuytelaars and Schmid, 2007]. Thus, density estimation is usually employed to find the densely occupied regions.

Figure 6.1 illustrates the role of each step employed in Bag-of-Words. Formulations for mid-level coding and pooling will now be described. Let us assume descriptor vectors $\boldsymbol{x}_n \in \mathbb{R}^D$ such that $n = 1, ..., N$, where N is the total descriptor cardinality for the entire image set $\mathcal{I}$, and D is the descriptor dimensionality. Further, $\mathcal{X} = \{\boldsymbol{x}_n\}_{n=1}^{N}$ can be viewed as a descriptor set or a matrix $\mathcal{X} \in \mathbb{R}^{D \times N}$ with the descriptors as column vectors. Given any image $i \in \mathcal{I}$, $\mathcal{N}^i$ denotes a set of its descriptor indices. We drop the superscript for simplicity and use $\mathcal{N}$. Next, let us assume we have $k = 1, ..., K$ visual appearance prototypes $\boldsymbol{m}_k \in \mathbb{R}^D$ a.k.a. visual vocabulary, words, centres, atoms, and anchors. We form a dictionary $\mathcal{M} = \{\boldsymbol{m}_k\}_{k=1}^{K}$ such that $\mathcal{M} \in \mathbb{R}^{D \times K}$. Additionally, if applied, $q = 1, ..., Q$ denotes partitions of a chosen Pyramid Matching, *e.g.* SPM from [Lazebnik et al., 2006, Yang et al., 2009], DoPM, or CoPM from chapter 5. It follows $\mathcal{N}_q^i \subseteq \mathcal{N}^i$ (we write $\mathcal{N}_q$ for simplicity) is a subset of the descriptor indices that fall into a given pyramid partition q of image i. Following the formalism of [Boureau et al., 2010a], we express the mid-level coding and pooling steps in BoW as:

$$\boldsymbol{\phi}_n = f(\boldsymbol{x}_n, \mathcal{M}), \quad \forall n \in \mathcal{N} \tag{6.1}$$

$$\psi_{kq} = g\left(\{\phi_{kn}\}_{n \in \mathcal{N}_q}\right), \quad \forall q = 1, ..., Q \tag{6.2}$$

$$\boldsymbol{h} = \hat{\boldsymbol{h}} / \|\hat{\boldsymbol{h}}\|_2 \ , \quad \hat{\boldsymbol{h}} = \left[\boldsymbol{\psi}_1{}^T, ..., \boldsymbol{\psi}_Q{}^T\right]^T \tag{6.3}$$

Equation (6.1) represents a chosen mid-level feature mapping $f : \mathbb{R}^D \to \mathbb{R}^K$, *e.g.* Soft Assignment or Sparse Coding. It quantifies the image content in terms of the visual prototypes given in $\mathcal{M}$. Each descriptor $\boldsymbol{x}_n$ is embedded into the visual vocabulary space resulting in mid-level features $\boldsymbol{\phi}_n \in \mathbb{R}^K$. In the following, we often refer to an n^{th} vector $\boldsymbol{\phi}_n$ or directly to a k^{th} coefficient of an n^{th} vector ϕ_{kn}. One can also think of

vectors $\boldsymbol{\phi}_n$ forming columns of matrix $\boldsymbol{\Phi}$ such that $\boldsymbol{\Phi} \in \mathbb{R}^{K \times |\mathcal{N}|}$. Note that $\mathcal{M}$ is formed from k-means cluster centres, later used by all mid-level coding approaches. Thus, equation (6.1) does not include the dictionary learning step. Figure 6.1 (a) illustrates descriptors $\{\boldsymbol{x}_n\}_{n \in \mathcal{N}}$ of image i, used by the coding step in figure 6.1 (b). Next, coding operates on each descriptor and produces corresponding mid-level features $\{\boldsymbol{\phi}_n\}_{n \in \mathcal{N}}$.

Equation (6.2) represents the pooling operation, *e.g.* Average or Max-pooling. The role of g is to aggregate occurrences of visual words in an image. Formally, function $g : \mathbb{R}^{|\mathcal{N}|} \rightarrow \mathbb{R}$ takes all mid-level feature coefficients ϕ_{kn} for visual word $\boldsymbol{m}_k$ given partition q of image i, and stores a value as a k^{th} coefficient in a q^{th} vector $\boldsymbol{\psi}_q \in \mathbb{R}^K$, denoted as ψ_{kq}. Moreover, one can think of vectors $\boldsymbol{\psi}_q$ as forming columns of matrix $\boldsymbol{\Psi}$ such that $\boldsymbol{\Psi} \in \mathbb{R}^{K \times Q}$. Figure 6.1 (c) depicts mid-level feature coefficients $\{\phi_{kn}\}_{n \in \mathcal{N}_q}$ which are used by the pooling step given $k = 1, ..., K$. Note that g acts on a given k^{th} row of mid-level features by aggregating occurrences of $\boldsymbol{m}_k$ into a k^{th} coefficient in $\boldsymbol{\psi}_q$.

Equation (6.3) concatenates $\boldsymbol{\psi}_q$ for all partitions $q = 1, ..., Q$ into $\hat{\boldsymbol{h}} \in \mathbb{R}^{KQ}$. It also normalises signature $\hat{\boldsymbol{h}}$ to preserve only relative statistics of visual word occurrences in an image, irrespective of the number of descriptors contained within it. This yields the final signature $\boldsymbol{h} \in \mathbb{R}^{KQ}$ of unit length as illustrated in figure 6.1 (d). The resulting signatures $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^{KQ}$ for $i, j \in \mathcal{I}$ can be directly fed to a primary-formulated SVM classifier or used to form a linear kernel $Ker_{ij} = (\boldsymbol{h}_i)^T \cdot \boldsymbol{h}_j$. This defines the similarity between images for kernel based classifiers such as KDA, latter used in this work.

The HA, SA, SC, LLC, and LcSA coding methods will now be described using the terms introduced above. For simplicity, $\boldsymbol{x}_n$ is referred to as $\boldsymbol{x}$, $\boldsymbol{\phi}_n$ as $\boldsymbol{\phi}$, and $\boldsymbol{\psi}_q$ as $\boldsymbol{\psi}$ where possible. Therefore, the notation for coefficients ϕ_{kn} and ψ_{kq} is further simplified to ϕ_k and ψ_k, respectively. Furthermore, we define the activation of anchor $\boldsymbol{m}_k$ given $\boldsymbol{x}$ as a response $\phi_k \neq 0$ and the local activation as $\phi_k \neq 0$ such that $r^2 = \|\boldsymbol{m}_k - \boldsymbol{x}\|_2^2$ and $r^2 < \kappa$ for an arbitrarily chosen constant $\kappa > 0$, where k defines a neighbourhood such that any two descriptors chosen from it have close visual appearances. Intuitively, $\phi_k \neq 0$ and $r^2 \geq \kappa$ define a non-local activation.

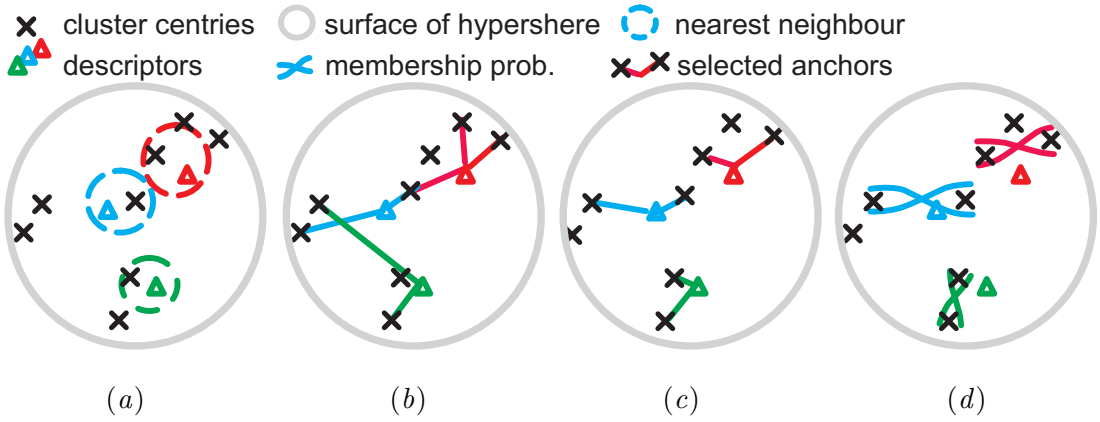


Figure 6.2: Illustration of (a) Hard Assignment, (b) Sparse Coding, (c) Locality-constrained Linear Coding, (d) Approximate Locality-constrained Soft Assignment. Descriptor vectors (triangles) are scattered on a surface of a hypersphere amongst the anchors (crosses). Note the difference between SC and LLC.

6.2.1 Hard Quantisation a.k.a. Hard Assignment (HA)

Bag-of-Words in its simplest form employs HA that solves the following problem:

$$\begin{aligned} \boldsymbol{\phi} &= \arg\min_{\bar{\boldsymbol{\phi}}} \left\| \boldsymbol{x} - \mathcal{M}\bar{\boldsymbol{\phi}} \right\|_2^2 \\ s.\,t.\ \ &\|\bar{\boldsymbol{\phi}}\|_1 = 1, \bar{\boldsymbol{\phi}} \in \{0,1\}^K \end{aligned} \tag{6.4}$$

In practice, equation (6.4) means that having formed a dictionary $\mathcal{M}$ by k-means clustering (or any other method), every descriptor $\boldsymbol{x} \in \mathcal{X}$ is assigned to its nearest cluster with activation equal 1. This is illustrated in figure 6.2 (a). The ℓ_1 norm constraint $\|\boldsymbol{\phi}\|_1 = 1$ ensures that $\boldsymbol{\phi}$ are histograms. Since $\boldsymbol{\phi}$ can take only binary values, the ℓ_1 norm also ensures a single non-zero entry per $\boldsymbol{\phi}$. Recently, it was shown that HA with appropriate pooling can achieve improved results [Boureau et al., 2010b, Chatfield et al., 2011] despite its inherently high quantisation error and largely compromised smoothness [Yu et al., 2009]. However, methods like Sparse Coding were shown to consistently perform significantly better. Therefore, we omit HA in the following evaluations.

6.2.2 Soft Assignment (SA)

The Soft Assignment coder is already introduced in section 4.2 of chapter 4. This approach is derived from Gaussian Mixture Model. Let us remind that K denotes

the number of visual words in a given dictionary, $\boldsymbol{m}_k$ are the visual words such that $k=1,...,K$, σ is the smoothing parameter of kernel G, and $\mathcal{X}=\{\boldsymbol{x}_n\}_{n=1}^{N}$ are descriptors of a dataset. Given the simplified density estimation problem in equation (4.3), the SA coder is equivalent to the membership probability of component k being selected given descriptor $\boldsymbol{x}$. We employ equation (4.4) and define SA as follows:

$$\phi_k = p(k|\boldsymbol{x}, \sigma) \tag{6.5}$$

Moreover, defining $\psi_k = \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \phi_{kn}$, where $\phi_{kn} = p(k|\boldsymbol{x}_n, \sigma)$, turns such a formulation into Visual Word Uncertainty from [van Gemert et al., 2010].

6.2.3 Sparse Coding (SC)

The goal of Sparse Coding [Lee et al., 2007, Yang et al., 2009] is to express each descriptor vector $\boldsymbol{x}$ as a sparse linear combination of the visual words given by $\mathcal{M}$. This can be achieved by optimising the following with respect to $\boldsymbol{\phi}$:

$$\begin{aligned} &\boldsymbol{\phi} = \arg\min_{\bar{\boldsymbol{\phi}}} \left\| \boldsymbol{x} - \mathcal{M}\bar{\boldsymbol{\phi}} \right\|_2^2 + \alpha \|\bar{\boldsymbol{\phi}}\|_1 \\ &s.\,t.\ \ \bar{\boldsymbol{\phi}} \geq 0 \end{aligned} \tag{6.6}$$

The ℓ_1 norm over $\boldsymbol{\phi}$ induces a low number of activations per descriptor, referred to as sparsity, which can be adjusted with α. SC was found to perform well if combined with Max-pooling and Spatial Pyramid Matching in [Yang et al., 2009]. Defining $\psi_k = \max\left(\{\phi_{kn}\}_{n \in \mathcal{N}}\right)$ in equation (6.2) renders this model equivalent to Sparse Coding from [Yang et al., 2009] except for: i) a skipped dictionary learning step, ii) a non-negative constraint[2] on $\boldsymbol{\phi}$. The image signatures in [Yang et al., 2009] are twice as long due to pooling over positive and negative ϕ_{kn} respectively. It is shown later that neglecting negative activations has no detrimental impact on the classification performance. Figure 6.2 (b) shows that SC can activate non-local anchors.

[2] To impose $\boldsymbol{\phi} \geq 0$ on SC and LLC, we used LAR [Efron et al., 2004] solver from SPAMS [Mairal et al., 2010] and Quadratic Programming [MOSEK, 2012], respectively. However, ignoring constraint $\boldsymbol{\phi} \geq 0$ and correcting SC and LLC codes by $\phi_k := max(0, \phi_k)$ for $k=1,...,K$ yielded equally good results.

6.2.4 Approximate Locality-constrained Linear Coding (LLC)

Locality-constrained Linear Coding [Wang et al., 2010] addresses the non-locality that can occur in Sparse Coding. It prevents activations of visual words that are far from descriptors. See figures 6.2 (b and c) for intuitive differences. This is formulated as:

$$\begin{aligned} \boldsymbol{\phi} &= \arg\min_{\bar{\boldsymbol{\phi}}} \left\| \boldsymbol{x} - \mathcal{M}\bar{\boldsymbol{\phi}} \right\|_2^2 + \alpha \sum_{k=1}^{K} \left(\bar{\phi}_k \cdot e^{\frac{\|\boldsymbol{x}-\boldsymbol{m}_k\|_2}{\sigma}} \right)^2 \\ &s.\,t.\ \ \mathbf{1}^T \bar{\boldsymbol{\phi}} = 1 \end{aligned} \tag{6.7}$$

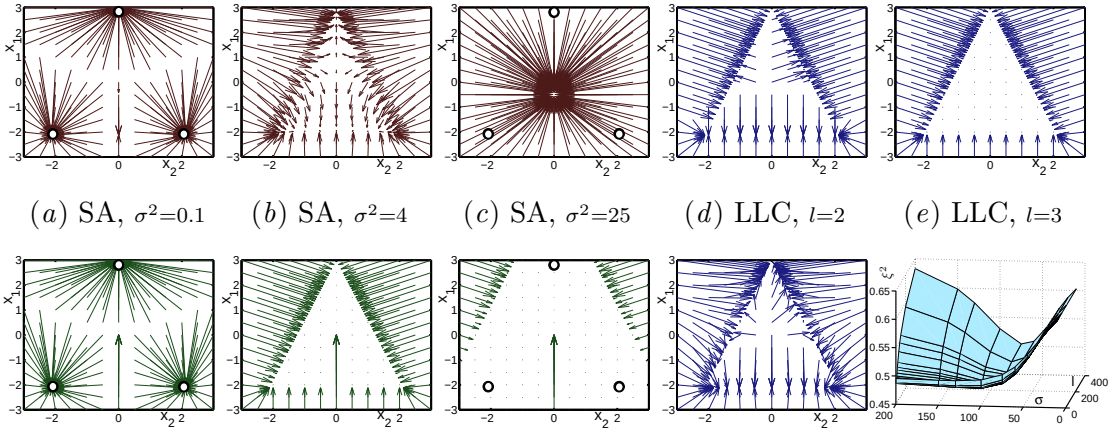
The squared ℓ_2 norm, expressed as a summation on the right side of equation (6.7), penalises large ϕ_k if the corresponding $\boldsymbol{m}_k$ is far from a given descriptor $\boldsymbol{x}$. The penalty can be adjusted by α and σ. This problem is equivalent to the formulation from [Wang et al., 2010], except for the dictionary learning step. In practice, we solve a fast approximate formulation:

$$\begin{aligned} \boldsymbol{\phi}^* &= \arg\min_{\bar{\boldsymbol{\phi}}} \left\| \boldsymbol{x} - \mathcal{M}\left(\boldsymbol{x}, l\right) \bar{\boldsymbol{\phi}} \right\|_2^2 \\ &s.\,t.\ \ \bar{\boldsymbol{\phi}} \geq 0,\ \ \mathbf{1}^T \bar{\boldsymbol{\phi}} = 1 \end{aligned} \tag{6.8}$$

Descriptor $\boldsymbol{x}$ is coded with its l-nearest neighbour anchors found in dictionary $\mathcal{M}$ by NN search, a new compact dictionary is formed and used: $\mathcal{M}\left(\boldsymbol{x}, l\right) = NN_{\mathcal{M}}\left(\boldsymbol{x}, l\right) \in \mathbb{R}^{D \times l}$, where $l \ll K$. Hence, one has to adjust l instead of α and σ. Note, the resulting $\boldsymbol{\phi}^* \in \mathbb{R}^l$ has length l. In practice, we re-project its elements into the full length vector $\boldsymbol{\phi} \in \mathbb{R}^K$ as, for each atom in $\mathcal{M}\left(\boldsymbol{x}, l\right)$, we know its position in $\mathcal{M}$. A non-negativity constraint[2] is applied to $\boldsymbol{\phi}$ as no classification improvement is observed if $\boldsymbol{\phi} < 0$ is allowed. Figure 6.2 (c) depicts a local selection of anchors for LLC.

6.2.5 Approximate Locality-constrained Soft Assignment (LcSA)

Sparse Coding from [Lee et al., 2007, Yang et al., 2009] and Locality-constrained Linear Coding from [Wang et al., 2010] are robust approaches that can learn a data manifold by approximating it with sparse and local linear combinations of anchors, respectively. This is achieved by constraining activations to a relevant subset of anchors. Thus, we constrain SA to activate only the l-nearest anchors of the descriptors as in [Wang et al., 2010, Lingqiao et al., 2011] when computing the membership probabilities. This



(*a*) SA, σ^2=0.1 (*b*) SA, σ^2=4 (*c*) SA, σ^2=25 (*d*) LLC, l=2 (*e*) LLC, l=3

(*f*) SC, α=100 (*g*) SC, α=1 (*h*) SC, α=0.5 (*i*) LcSA, σ^2=4, l=2 (*j*) ξ^2 for LcSA

Figure 6.3: The quantisation error: flow of the descriptors from their original positions $\boldsymbol{x}$ denoted by the grid points to the corresponding reconstructed positions pointed to by the arrows. (*a*) SA: the descriptors are moved to their nearest anchors '$\circ$' like in HA. (*b*) SA: a near-optimal smoothing factor case yielding low ξ^2. (*c*) SA: a full blur of the data for large σ. The reconstructed positions overlap in the centre. (*d*) LLC: limited reconstruction due to low $l=2$. (*e*) LLC: optimal reconstruction within the triangular region given $l=3$. (*f*) SC: the descriptors are moved to their nearest anchors '$\circ$' like in HA. Note, $\|\boldsymbol{\phi}\|_1 = 1/\alpha$ had to be rescaled to $\|\boldsymbol{\phi}\|_1 = 1$ to prepare this plot. (*g*) SC: optimal reconstruction within the triangular region. (*h*) SC: area of the optimal reconstruction is increased for small α at a price of non-sparsity. (*i*) LcSA: reconstruction capabilities of LcSA resemble closely LLC case (d). (*j*) LcSA: cost ξ^2 resulting from combining equations (6.9) and (4.5), shown as a function of (σ, l).

is illustrated in figure 6.2 (d). This is referred to as Approximate Locality-constrained Soft Assignment. Recall that $\mathcal{M}(\boldsymbol{x}, l) = NN_{\mathcal{M}}(\boldsymbol{x}, l) \in \mathbb{R}^{D \times l}$ is a set of the l-nearest anchors of descriptor $\boldsymbol{x}$ given dictionary $\mathcal{M}$ such that $l \ll K$. Limiting the membership probability from equation (4.4) to be spanned with only l-local anchors $\mathcal{M}(\boldsymbol{x}, l)$ yields:

$$\phi_k = p(k|\boldsymbol{x}, \sigma, l) = \begin{cases} \frac{G(\boldsymbol{x}; \boldsymbol{m}_k, \sigma)}{\sum_{\boldsymbol{m}' \in \mathcal{M}(\boldsymbol{x}, l)} G(\boldsymbol{x}; \boldsymbol{m}', \sigma)} & \text{if } \boldsymbol{m}_k \in \mathcal{M}(\boldsymbol{x}, l) \\ 0 & \text{otherwise} \end{cases} \tag{6.9}$$

Moreover, appendix A.1 demonstrates the analytical similarity between LcSA and LLC.

6.2.6 Mid-level Coding Parameters

To achieve good performance, SC and LLC optimise a trade-off between a quantisation loss (defined below) and an explicitly chosen regularisation penalty, *e.g.* sparsity as in equation (6.6) or locality as in equation (6.7). Such a trade-off can be subjected to additional constraints, *e.g.* non-negativity and an upper limit on the solution. The quality of quantisation in these mappings is measured in accordance with the theory of Linear Coordinate Coding [Yu et al., 2009] also described in section 4.3 of chapter 4.

We know from equation (4.5) provided in section 4.3 that transforming descriptor $\boldsymbol{x}$ into mid-level feature $\boldsymbol{\phi} = f(\boldsymbol{x})$ results in a quantisation loss $\xi^2(\boldsymbol{x})$ a.k.a. the residual error which depends on the choice of mapping f. Transforming the mid-level feature back into the descriptor yields $\xi^2(\boldsymbol{x})$. The approximation error of N descriptors is $\xi^2 = \frac{1}{N}\sum_n \xi^2(\boldsymbol{x}_n)$. We assume ξ^2 is synonymous with the quantisation error, which is a source of ambiguity in the coding methods, *e.g.* HA or SA.

Moreover, regularisation terms must be imposed on this least squares problem employed by the LCC family to ensure that each descriptor is coded by a representative fraction of atoms. For instance, we observed with the SC and LLC coders that given the optimal regularisation parameters, mid-level features from various classes of textures exhibit high intra-class and low inter-class similarity. However, removing regularisation leads to a sharp increase of inter-class similarity. Such mid-level features are not distinctive enough for a pooling step to produce informative signatures.

Figure 6.3 presents how mid-level features are affected by the quantisation error. Having coded descriptors $\boldsymbol{x} = [x_1, x_2]^T \in \langle -3; 3\rangle^2$ with $k = 1, 2, 3$ atoms $\boldsymbol{m}_k$ by various methods, the obtained codes $\boldsymbol{\phi}$ are projected back to the descriptor space: $\hat{\boldsymbol{x}} = \mathcal{M}\boldsymbol{\phi}$. The resulting quantisation effects are visualised as displacements between each descriptor $\boldsymbol{x}$ and its approximation $\hat{\boldsymbol{x}}$. Plots (a-c) present SA with low σ (HA equivalent), optimal, and large σ (data blur: if $\sigma \to +\infty$, then $\phi_k \to 1/K$). Plot (d) shows LLC, which modifies the descriptor space for $l = 2$. Plot (e) shows LLC yielding a good reconstruction for $l = 3$, however, this causes non-locality. Plots (f-h) show SC with high α (HA equivalent, $\|\boldsymbol{\phi}\|_1 = 1/\alpha$ was rescaled to $\|\boldsymbol{\phi}\|_1 = 1$), medium α (good trade-off), and low α at a price of non-sparsity. Plot (i) shows LcSA approximating LLC in plot (d). Lastly, plot 6.3

(j) shows the ξ^2 cost for LcSA coder f in equation (6.9) as a function of (σ, l) yielded by equation (4.5). Note, $\xi^2 > 0$ has a unique minimum and it varies smoothly with changes of (σ, l). Various descriptors and datasets consistently resulted in a unique minimum. Appendix A.5 illustrates the activation spaces spanned by the coding methods.

Typically, the optimal coding parameters are determined during the cross-validation process. We found empirically that minimising $\xi^2 > 0$ w.r.t. (σ, l) in the LcSA model led to good classification results. This can be explained by two trade-off factors: i) Extreme σ results in either HA or the data blur as shown in plots 6.3 (a-c). Thus, measuring ξ^2 can be used to penalise selection of such extremes. ii) Usually, given the ℓ_2 norm normalised data, descriptor $\boldsymbol{x}$ coded with the distant anchors yields approximation $\hat{\boldsymbol{x}}_1$ such that $\|\hat{\boldsymbol{x}}_1\|_2 < \|\boldsymbol{x}\|_2$ due to various implicit constraints of LcSA, *e.g.* $\boldsymbol{\phi} \geq 0$, $\|\boldsymbol{\phi}\|_1 = 1$. However, coding $\boldsymbol{x}$ with both distant and nearby anchors yields $\hat{\boldsymbol{x}}_2$ such that $\|\hat{\boldsymbol{x}}_1\|_2 < \|\hat{\boldsymbol{x}}_2\|_2 < \|\boldsymbol{x}\|_2$. Lastly, coding $\boldsymbol{x}$ with its nearby anchors only yields $\hat{\boldsymbol{x}}_3$ such that $\|\hat{\boldsymbol{x}}_1\|_2 < \|\hat{\boldsymbol{x}}_2\|_2 < \|\hat{\boldsymbol{x}}_3\|_2 < \|\boldsymbol{x}\|_2$. This suggests ξ^2 shown in plot 6.3 (j) favours local coding in LcSA. Thus, we combine equations (6.9) and (4.5) to find the initial σ and l-nearest anchors:

$$(\sigma,\ l) = \underset{(\bar{\sigma},\ \bar{l})}{\arg\min} \sum_{n=1}^{N} \left\| \boldsymbol{x}_n - \sum_{\boldsymbol{m} \in \mathcal{M}(\boldsymbol{x}_n, \bar{l})} \frac{G(\boldsymbol{x}_n; \boldsymbol{m}, \bar{\sigma})}{\sum_{\boldsymbol{m}' \in \mathcal{M}(\boldsymbol{x}_n, \bar{l},)} G(\boldsymbol{x}_n; \boldsymbol{m}', \bar{\sigma})} \cdot \boldsymbol{m} \right\|_2^2 \tag{6.10}$$

Such evaluated parameters were found to provide good initial estimates. Next, (σ, l) can be adjusted by cross-validation for optimal classification performance. Similar approach demonstrated good empirical results for SA in chapter 4. Appendix A.2 explains how to efficiently optimise the cost in equation (6.10) in order to find parameters $(\sigma,\ l)$.

6.2.7 Computational Efficiency

When embedding descriptors (*e.g.* $6K$ per image) of a medium scale dataset to a vocabulary space (*e.g.* $16K$ atoms), the computational cost of coding becomes a major factor in experiments. Thus, this section details the computational complexity of HA, SA, LcSA, SC, and LLC and proposes an approach which increases the speed of coding.

HA. It requires the NN search which scales linearly with the number of descriptors N and the number of visual words K. This results in a complexity $\mathcal{O}(N \times K)$.

SA. Soft Assignment computes: i) Gaussian-based distances from a descriptor to each visual word, ii) the sum of such distances, iii) the ratio of (i) to the total distance (ii) as in equation (4.4). Therefore, $\mathcal{O}(N\times 3K)=\mathcal{O}(N\times K)$.

SC. The complexity of Sparse Coding based on the Feature Sign [Lee et al., 2007] solver is expressed as $\mathcal{O}(N\times K\times S)$, where S is the average number of non-zero elements in the mid-level features. The complexity of the Least Angle Regression [Efron et al., 2004] based solver proposed in [Mairal et al., 2010] is $\mathcal{O}(N\times S^3 + N\times K\times S^2 + N\times K\times S)=\mathcal{O}(N\times K\times S^2)$ for $S \ll K$.

LLC. Because Locality-constrained Linear Coding is $\mathcal{O}(N\times K^2)$ complex, Approximate LLC was also introduced in [Wang et al., 2010]. It has a more favourable complexity $\mathcal{O}(N\times K\times log\ l + N\times l^2)=\mathcal{O}(N\times K\times log\ l)$ for $l \ll K$ nearest anchors.

LcSA. The speed of Approximate Locality-constrained Soft Assignment is restricted by the NN search based on the partial sort algorithm with complexity $\mathcal{O}(N\times K\times log\ l)$, where l is the number of nearest anchors in the search. Summing distances and computing the ratio of Gaussians in equation (6.9) becomes an efficient task with complexity $\mathcal{O}(N\times 2l)$. Hence, the total complexity is $\mathcal{O}(N\times K\times log\ l + N\times 2l) = \mathcal{O}(N\times K\times log\ l)$. Note that LcSA becomes noticeably faster than SA for $log\ l \ll 3$ since $N\times K\times log\ l \ll N\times 3K$.

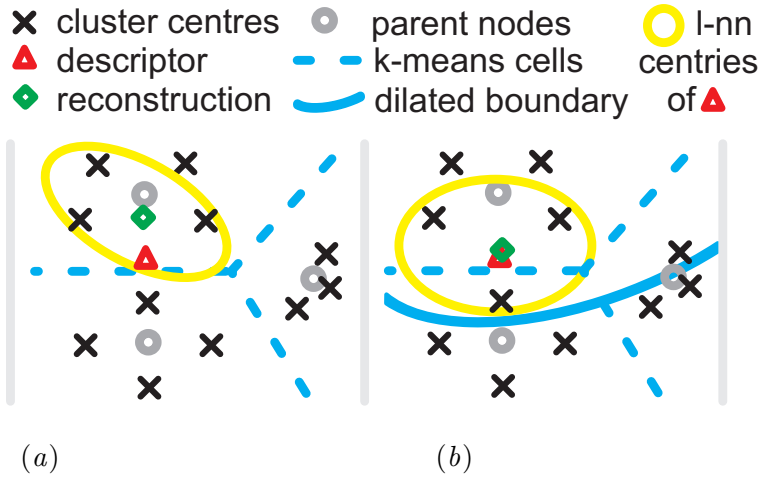


Figure 6.4: (a) Hierarchical NN: l-nearest anchors of a descriptor found in its nearest k-means cluster. (b) Dilating cluster boundaries improves quantisation: a descriptor and its reconstruction are brought closer.

FHNNS. To increase coding speed, we propose a Fast Hierarchical Nearest Neighbour Search method that uses an approximate dictionary search for the l-nearest neighbours of a to-be-coded descriptor $\boldsymbol{x}$ to form a compact dictionary $\mathcal{M}(\boldsymbol{x}, l)$. Figure 6.4 (a) shows a hierarchical k-means vocabulary with two levels of depth. The parent node which is closest to $\boldsymbol{x}$ is found and then the l-nearest children. However, such a process results in a high quantisation jitter and a poor selection of anchors. Thus, we propose to share k-means children nodes located along boundaries between their parent nodes. The dilation of k-means boundaries is shown in figure 6.4 (b). A similar approach to NN search is used by Spill Trees [Liu et al., 2004]. To measure the corresponding quantisation noise the formula (4.5) from chapter 4 is used over a set of descriptors.

In detail, for every k-means parent node $\boldsymbol{m'} \in \mathcal{M}'$ its dilated set of children $\hat{\mathcal{M}}(\boldsymbol{m'}, \ell)$ is defined as $\hat{\mathcal{M}}(\boldsymbol{m'}, \ell) = NN_{\mathcal{M}}(\boldsymbol{m'}, \ell)$: the ℓ-nearest neighbours of each $\boldsymbol{m'}$ are chosen from the dictionary $\mathcal{M}$ representing the original child nodes of k-means. To increase the speed of LcSA and LLC, we combine two search operations such that $\boldsymbol{m'} = NN_{\mathcal{M}'}(\boldsymbol{x}, 1)$ indicates the nearest parent node $\boldsymbol{m'}$ of $\boldsymbol{x}$ and $\mathcal{M}(\boldsymbol{x}, l) = NN_{\hat{\mathcal{M}}(\boldsymbol{m'}, \ell)}(\boldsymbol{x}, l)$ forms a compact dictionary for $\boldsymbol{x}$. For SC, we take the nearest parent node $\boldsymbol{m'}$ of $\boldsymbol{x}$ and code $\boldsymbol{x}$ using the dilated dictionary $\hat{\mathcal{M}}(\boldsymbol{m'}, \ell)$. Varying $\ell = 1, ..., K$ affects a trade-off between speed and accuracy. In all cases, mid-level features remain of length K, rather than ℓ,

4K/128D

SA	LcSA	LLC	SC
2.26	0.24	0.44	3.61
LcSA ℓ=256	LcSA ℓ=512	LcSA ℓ=1024	LcSA ℓ=2048
0.036	0.046	0.074	0.136

16K/192D

SA	LcSA	LLC	SC
13.8	1.06	1.55	32.5
SC ℓ=1024	SC ℓ=2048	SC ℓ=3072	SC ℓ=4096
3.69	8.74	14.7	21.8

Table 6.1: Computational time (in seconds) required to code $1K$ SIFT descriptors to mid-level features. (*Top*) $4K$ dictionary and $128D$ descriptors. (*Bottom*) $16K$ dictionary and $192D$ descriptors.

as we re-project them for each atom in $\hat{\mathcal{M}}(\boldsymbol{m'}, \ell)$ to its corresponding position in $\mathcal{M}$. The complexity of LcSA and LLC becomes $\mathcal{O}(N \times \ell_p + N \times \ell \times log\ l) = \mathcal{O}(N \times \ell \times log\ l)$, for $\ell_p \ll \ell \ll K$ and $l \ll \ell$, where ℓ_p and ℓ are a number of parent nodes[3] and children per node, respectively. The complexity of SC is thus $\mathcal{O}(N \times \ell \times S^2)$.

Timing. Table 6.1 shows the computation times on a single 2.3GHz AMD Opteron core that are required to code $1K$ SIFT descriptors of 128 and 192 dimensions to mid-level features for $4K$ and $16K$ dictionaries. LcSA can run 4 times faster without a loss in its classification performance, as shown in section 6.4.3. SC also gains on speed.

6.3 Overview of Pooling Approaches

Pooling converts mid-level features into final image signatures by aggregating occurrences of visual words in each image. Formally, equation (6.2) expresses its place in the context of Bag-of-Words. Pooling is performed in each pyramid partition q of image i, $\mathcal{N}_q^i$ denotes a subset of descriptor indices to be processed. We abbreviate $\mathcal{N}_q^i$ to $\mathcal{N}$ and $\boldsymbol{\psi}_q$ to $\boldsymbol{\psi}$ for clarity. Moreover, the notation for coefficients ψ_{kq} is further simplified to ψ_k. Lastly, we refer to ϕ_{kn} as a k^{th} coefficient of an n^{th} vector $\boldsymbol{\phi}_n$.

6.3.1 Average (Avg), Max-pooling (Max), Mix-order Max-pooling (MixOrd), and an ℓ_p norm based trade-off (lp-norm)

Average and Max-pooling are intuitively introduced in section 6.1 and referred to in sections 6.2.2 and 6.2.3. To summarise, Average pooling is expressed as the average over responses to visual word $\boldsymbol{m}_k$:

$$\psi_k = \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \phi_{kn} \tag{6.11}$$

Maximum pooling intuitively selects the largest value between mid-level features responding to visual word $\boldsymbol{m}_k$:

$$\psi_k = \max\left(\{\phi_{kn}\}_{n \in \mathcal{N}}\right) \tag{6.12}$$

[3]Note that the ℓ_p symbol used in this context is not an ℓ_p norm.

Therefore, the fundamental difference is that Average pooling counts all occurrences of visual word $\boldsymbol{m}_k$ in the image while Max-pooling only registers a presence of $\boldsymbol{m}_k$. Max-pooling has been shown to be a lower bound of the likelihood of *at least one visual word* $\boldsymbol{m}_k$ *being present in image* i [Lingqiao et al., 2011]. This however does not clarify whether the lower bound formulation is more suited for classification than the exact analytical solution.

Further, Mix-order Max-pooling is proposed in [Lingqiao et al., 2011] as a lower bound of *at least* s *visual words* $\boldsymbol{m}_k$ *being present in image* i. This is achieved by sorting all mid-level feature entries corresponding to a visual word $\boldsymbol{m}_k$ and selecting exactly the s^{th} largest value. This process is performed for $k=1,...,K$ and it results in an image signature. Furthermore, selecting t different values of s (*e.g.* $s_1 > s_2 > ... > s_t$) yields t different image signatures per image. They form separate kernels that can be combined using kernel methods [Lingqiao et al., 2011].

Lastly, a trade-off between Average and Max-pooling was proposed in [Boureau et al., 2010b]. It employs an ℓ_p norm with parameter p which varies the solution between Average and Max-pooling for $p=1$ and $p \to \infty$, respectively:

$$\psi_k = \left(\frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} |\phi_{kn}|^p \right)^{1/p} \tag{6.13}$$

6.3.2 Theoretical expectation of Max-pooling (MaxExp) and at least one visual word $\boldsymbol{m}_k$ present in image i (ExaPro)

Likelihood based pooling methods have recently shed new light on the role of the pooling step in Bag-of-Words. It was shown in [Boureau et al., 2010b] that Max-pooling can be predicted analytically by drawing mid-level features (for a chosen $\boldsymbol{m}_k$) from Bernoulli distribution under the i.i.d. assumption. We assume the probability p for an event $(\phi_{kn}=1)$ and $1-p$ for $(\phi_{kn}=0)$. Probability of all $\bar{N} = |\mathcal{N}|$ mid-level features to be $\{(\phi_{k1}=0), ..., (\phi_{k\bar{N}}=0)\}$ amounts to $(1-p)^{\bar{N}}$. Similarly, the probability of at least one mid-level feature event $(\phi_{kn}=1)$ can be thought of as applying a logical 'or' operation

$\left\{(\phi_{k1}=1) \,\middle|\, ... \,\middle|\, (\phi_{k\bar{N}}=1)\right\}$ and is defined as:

$$\sum_{n=1}^{\bar{N}} \binom{\bar{N}}{n} p^n (1-p)^{\bar{N}-n} = 1-(1-p)^{\bar{N}} \tag{6.14}$$

Estimating p as the average of mid-level feature activations for a given $\boldsymbol{m}_k$ results in the final MaxExp formulation:

$$\psi_k = 1 - \left(1 - \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}} \phi_{kn}\right)^{\bar{N}}, \; \bar{N} = |\mathcal{N}| \tag{6.15}$$

Next, similar assumptions to MaxExp were taken in [Lingqiao et al., 2011]: mid-level features represent random variables drawn from a feature distribution under the i.i.d. assumption. Therefore, the probability of *at least one visual word* $\boldsymbol{m}_k$ *present in image* i (ExaPro) is defined as:

$$\psi_k = 1 - \prod_{n\in\mathcal{N}} (1-\phi_{kn}) \tag{6.16}$$

Note that the probabilistic interpretation of ExaPro also holds for MaxExp due to the way it acts on Average pooling. The next section shows that Power Normalisation used for Fisher Vector Encoding [Perronnin et al., 2010] acts similarly on Avg.

6.3.3 Power Normalisation a.k.a. Gamma Correction (Gamma)

Power Normalisation has been successfully applied to Intersection Kernels [Boughorbel et al., 2005], Fisher Vector Encoding [Perronnin et al., 2010], and in image retrieval [Jégou et al., 2009]. This is also known as Gamma Correction. Such a correction is shown to tackle *burstiness*: a phenomenon that a given visual word appears in an image more often than is statistically expected [Jégou et al., 2009]. Gamma acts on Average pooling to improve the similarity of the image signatures belonging to each class of objects and it is expressed as:

$$\psi_k = \left(\frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}} \phi_{kn}\right)^{\gamma} \tag{6.17}$$

The correction factor $0 < \gamma \leq 1$ is usually found by cross-validation. Note, setting $\gamma = 0.5$ changes a dot product between such formed vectors $\boldsymbol{\psi}$ into Bhattacharyya coefficient [Jebara et al., 2004]. As the nature of Gamma is not explored in previous

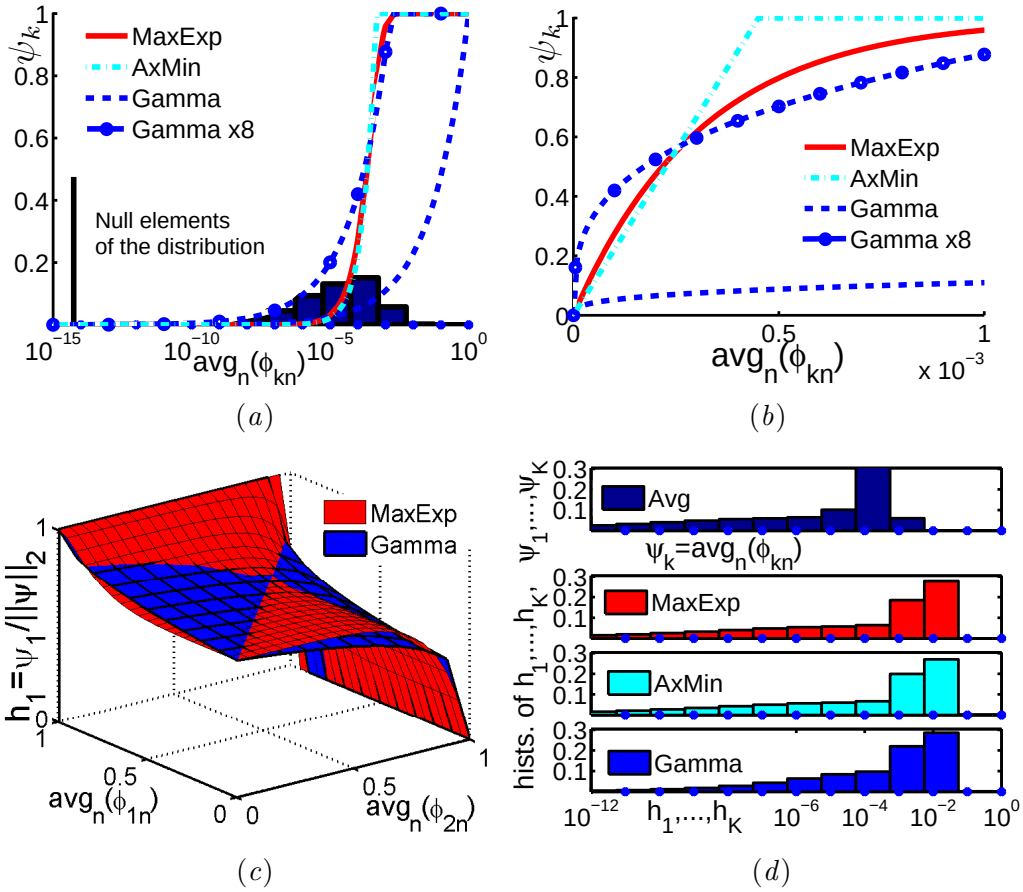


Figure 6.5: Illustration of the pooling correction functions: MaxExp, AxMin, and Gamma. (*a*) Bar plot is a histogram of Average pooling $avg_n(\phi_{kn})$ over $n=1,...,N$ for $k=1,...,K$ on Caltech101. AxMin and Gamma (if magnified $\times 8$) curves are approximations of MaxExp. Note the logarithmic scale. (*b*) Pooling methods as functions of Average pooling (linear scale). (*c*) ℓ_2 norm normalised MaxExp and Gamma as functions of Avg on a dictionary $K=2$ atoms (response h_1 for $\boldsymbol{m}_1$ is showed while we skip h_2 for clarity). (*d*) Histogram of Average pooling for $k=1,...,K$ on Flower17 is rearranged by MaxExp, AxMin, and Gamma, then the ℓ_2 norm normalised. This results in similar distributions (null entries not shown).

studies [Boughorbel et al., 2005, Perronnin et al., 2010, Jégou et al., 2009], our study found it closely related to MaxExp. According to equations (6.15) and (6.17), these two corrections are functions of Average pooling. Thus, the best performing correction curves were plotted on Caltech101 in figure 6.5 (a, b). Both MaxExp and Gamma $\times 8$ (magnified $\times 8$) have a similar appearance. They rapidly expand input intervals

$\langle 0; 0.0005\rangle$ and $\langle 0.0005; 0.001\rangle$ having equal lengths to output intervals $\langle 0; 0.8\rangle$ and $\langle 0.8; 0.98\rangle$ of two different lengths 0.8 and 0.18. Hence, the importance of low averages of activations increases when compared to the strong cases. The similarity of MaxExp and Gamma (not to be confused with Gamma ×8) becomes clear in figure 6.5 (c) due to the ℓ_2 norm normalisation as in equation (6.3). Averages of $2D$ mid-level features are used as the inputs for MaxExp and Gamma. Only γ is adjusted for the best fit between two curves. The resulting ℓ_2 norm normalised histogram bins $h_1 = \psi_1 / ||\boldsymbol{\psi}||_2$ are shown. With the ℓ_2 norm handling the scaling, MaxExp and Gamma become similar.

To validate whether Gamma and MaxExp act similarly in practice, a registration experiment was conducted. Assume $\hat{\boldsymbol{h}}_{\boldsymbol{i}}^{\boldsymbol{exp}}$ are known image signatures generated with MaxExp pooling for its known optimal $\bar{N}$, while $\hat{\boldsymbol{h}}_{\boldsymbol{i}}^{\gamma}$ are corresponding signatures generated with Gamma pooling for various candidates $\bar{\gamma}$. An unknown parameter γ of $\hat{\boldsymbol{h}}_{\boldsymbol{i}}^{\gamma}$ is sought that minimises the least squares error between image signatures of MaxExp and Gamma for images $i \in \mathcal{I}$:

$$\gamma = \arg\min_{\bar{\gamma}} \sum_{i \in \mathcal{I}} \left\| \frac{\hat{\boldsymbol{h}}_{\boldsymbol{i}}^{\boldsymbol{exp}}}{\|\hat{\boldsymbol{h}}_{\boldsymbol{i}}^{\boldsymbol{exp}}\|_2} - \frac{\hat{\boldsymbol{h}}_{\boldsymbol{i}}^{\bar{\gamma}}}{\|\hat{\boldsymbol{h}}_{\boldsymbol{i}}^{\bar{\gamma}}\|_2} \right\|_2^2 \tag{6.18}$$

Indeed, section 6.4.2 later shows that the best performing γ determined by cross-validation matches closely γ found by optimising the target in equation (6.18).

6.3.4 Modelling the Impact of Descriptor Interdependency on Analytical Pooling

The standard approach to Bag-of-Words typically assumes the descriptor extraction on a dense grid [van Gemert et al., 2008, 2010, Philbin et al., 2008, Lingqiao et al., 2011, Yang et al., 2009, Wang et al., 2010, Gao et al., 2010]. Thus, neighbouring descriptors largely overlap with each other. MaxExp and ExaPro pooling assume that activations ϕ_k of anchor $\boldsymbol{m}_k$ are independent in each image. However, if descriptor $\boldsymbol{x}$ results in activation ϕ_k of $\boldsymbol{m}_k$, descriptors significantly overlapping with $\boldsymbol{x}$ should also result in activations ϕ_k of $\boldsymbol{m}_k$. The same holds for repeatable visual patterns. Thus, we expect the average activation p (Average pooling) in equation (6.14) to be overestimated and p should be decreased by some factor μ, *e.g.* $p_{new} := (1-\mu)\, p$, where $0 \leq \mu < 1$. To

correct MaxExp, the parameter $\bar{N}$ in equation (6.15) is adjusted such that $1 \leq \bar{N} \leq |\mathcal{N}|$; this has the same effect as decreasing p. Gamma pooling can be corrected by varying γ or predicting it by equation (6.18) from the optimal $\bar{N}$ of MaxExp. In the next section, the descriptor interdependence is shown in a simulation, with an approach to take further advantage of it.

First, let us define a close approximation of MaxExp that has a parameter β accounting for the interdependence of descriptors. Approximate Pooling (AxMin) is expressed as:

$$\psi_k = \min\left(1,\ \beta p\right) = \min\left(1,\ \beta \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \phi_{kn}\right),\ \ 1 \leq \beta \leq |\mathcal{N}| \tag{6.19}$$

The AxMin curve, shown in figure 6.5 (a, b) on page 98, follows closely MaxExp and represents a linear magnifying function with a saturation threshold. It can be shown that the steepness β of AxMin and $\bar{N}$ of MaxExp are related such that $\beta \approx \bar{N}$. Parameters β and μ are related by $\beta = |\mathcal{N}|\,(1-\mu)$, hence adjusting β accounts for the interdependence of descriptors. AxMin pooling implies that the confidence in the visual word $\boldsymbol{m}_k$ being present in image i can increase until it reaches the saturation threshold (full confidence). Once reached, any strong variations have no effect which discards the noise. This also prevents the counting of any further occurrences of $\boldsymbol{m}_k$. Such a behaviour increases intra-class similarity of the image signatures and therefore resembles MaxExp and Gamma methods.

To summarise MaxExp, AxMin, and Gamma, figure 6.5 (d) on page 98 presents a distribution of coefficients of Average pooling on Flower17 by binning all ψ_k for $k = 1, ..., K$ for all images. Next, Average pooling is corrected with MaxExp, AxMin, and Gamma. The ℓ_2 norm normalisation is applied per image and all signature coefficients h_k are binned. The similar distributions of MaxExp, AxMin, and Gamma highlight their closeness as shown in sections 6.3.3 and 6.3.4.

6.3.5 Cross Vocabulary Leakage, Descriptor Interdependence, and Improved Pooling (@n)

To understand why Max-pooling is a solid performer despite it being merely a lower bound of *at least one visual word* $\boldsymbol{m}_k$ *present in image* i, the primary factors that can

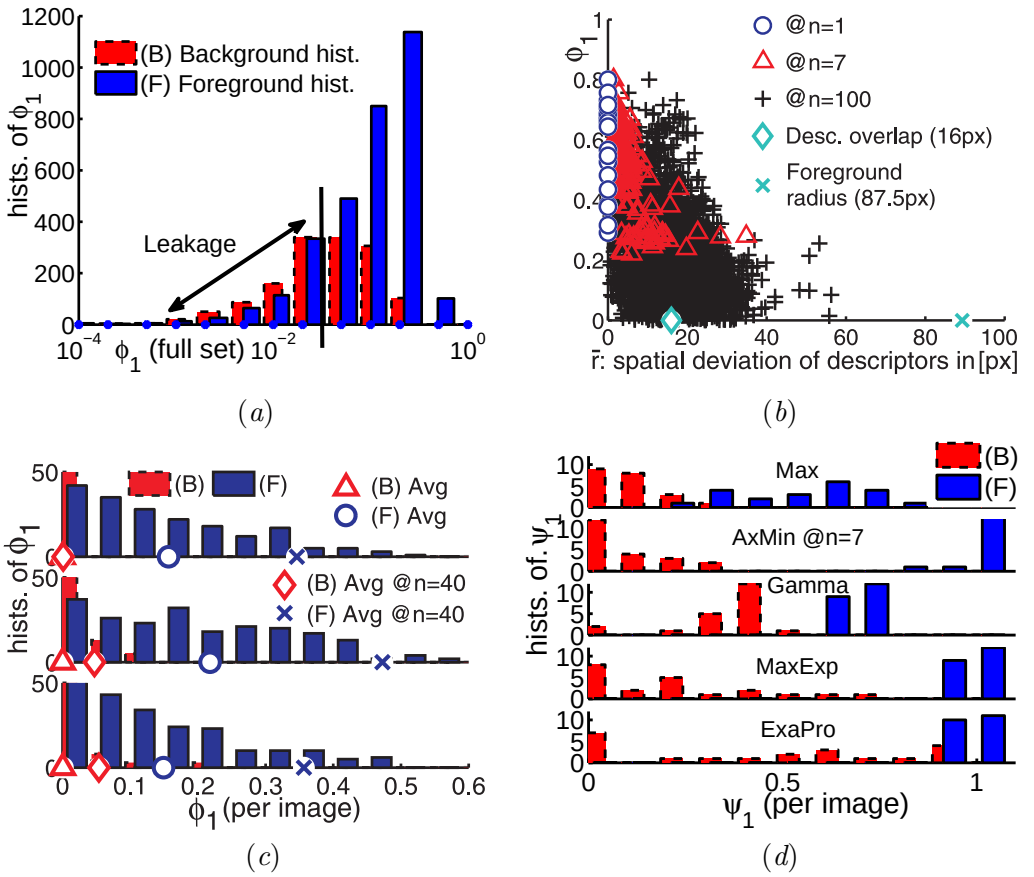


Figure 6.6: Toy experiment with 21/21 bounding boxes of faces/backgrounds. (*a*) Histograms of SC activations ϕ_1 for both foreground and background descriptors given visual word $\boldsymbol{m}_1$ that represents a nose. (*b*) Top 1, 7, and 100 largest activations ϕ_1 given $\boldsymbol{m}_1$ per foreground bounding box as functions of spatial deviation $\bar{r}$ between the descriptors inducing these activations. (*c*) 6 histograms of activations ϕ_1 given $\boldsymbol{m}_1$ for arbitrarily chosen 3 foreground and 3 background bounding boxes denoted as (F) and (B). Values of Average pooling are marked with circles and triangles, respectively, while Avg@$n = 40$ with crosses and diamonds. Note small separation distances between circles and triangles and large between crosses and diamonds. (*d*) Pooling methods are used to separate 21 faces from 21 backgrounds. Histograms of pooling responses ψ_1 (one ψ_1 per bounding box) given $\boldsymbol{m}_1$ are shown. Foreground and background are labelled as (F) and (B). Refer text for details.

affect pooling are discussed: i) cross vocabulary leakage, ii) propagated measurement error, iii) descriptor interdependence. These factors are addressed by an improved pool-

ing strategy called @n. Note, terms such as activation and local/non-local activation have been defined in section 6.2.

Leakage. Cross vocabulary leakage can be defined as activation $\phi_k \neq 0$ of visual word $\boldsymbol{m}_k$ given descriptor $\boldsymbol{x}$ that should not occur but it does due to: a) the inherent nature of a particular mid-level coding to trigger non-local activations, b) features not representing $\boldsymbol{m}_k$ but having visual appearances similar to $\boldsymbol{m}_k$, hence triggering ϕ_k. Leakage activation $\phi_k \neq 0$ may have an associated correct activation $\phi_{k'} \neq 0$ for $k \neq k'$, hence *cross vocabulary* terminology.

Soft Assignment is used to illustrate case (a). Let us assume descriptor $\boldsymbol{x}$ such that $\boldsymbol{x} = \boldsymbol{m}_k$. This results in activations not related to $\boldsymbol{m}_k$ because $p(k^*|\boldsymbol{x}, \sigma) > 0$ for any $\boldsymbol{m}_{k^*} \in \mathcal{M} \setminus \{\boldsymbol{m}_k\}$. Similar observations hold for $\boldsymbol{x} \neq \boldsymbol{m}_k$. SA results in large amounts of such a leakage, while LLC and LcSA circumvent this problem by suppressing most non-local activations explicitly in equations (6.8) and (6.9). Sparse Coding, however, allows non-local activations.

To illustrate leakage in SC, a toy experiment is introduced. 21 images of a subject's face were captured at similar scales and rotations, backgrounds varied. We applied SIFT (4px grid interval, 16px radii). Next, a descriptor from the first image centred at the tip of the subject's nose was selected. With 32×32 pixel area, it does not cover eyes, lips, or cheeks. It was added as the first element $\boldsymbol{m}_1$ to a dictionary of $4K$ k-means atoms trained on background images. Descriptors within manually annotated bounding boxes (160×190 pixel) of faces are deemed foreground samples. Further, 21 bounding boxes (160×190 pixel) were selected at random from backgrounds. Figure 6.6 (a) shows histograms of SC activations ϕ_1 for both foreground and background descriptors. Foregrounds tend to yield the majority of the large responses. Note that below a certain value of ϕ_1, indicated with a vertical bar, background descriptors respond to $\boldsymbol{m}_1$ more often than foreground descriptors. This shows the leakage case (a, b) in practice.

Propagation Error. Having formulated the leakage, the propagation error of MaxExp is computed w.r.t. the average activation $\phi_k = \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \phi_{kn}$ on its input. Applying the first derivative to eq. (6.15) w.r.t. ϕ_k and assuming a measurement uncertainty $\Delta\phi_k$ representing the leakage error leads to: $\Delta\psi_k = \Delta\phi_k \cdot \bar{N} \, (1 - \phi_k)^{\bar{N}-1}$. Let us assume $\bar{N}$

to be equal to the average count of descriptors per image, *e.g.* $\bar{N} = 6000$, and the leakage error $\Delta\phi_k = 10^{-5}$. For the sample means $\phi_k = 10^{-5}$ and $\phi_k = 10^{-4}$ the absolute propagation errors are $\Delta\psi_k = 0.056$ and $\Delta\psi_k = 0.032$ respectively. Larger $\Delta\psi_k$ given smaller ϕ_k suggests that MaxExp is sensitive to variations $\Delta\phi_k$ for small ϕ_k and can magnify small perturbations, *e.g.* the leakage. Equivalent findings apply to Gamma and ExaPro. Note that Max-pooling selects only the largest ϕ_{kn} over all $n \in \mathcal{N}$. Thus, it can suppress the leakage but it may be less robust to abrupt changes of large ϕ_{kn} when compared to analytical pooling. Hence, a compromise between Max-pooling and analytical methods is desired.

Descriptor Interdependence. Section 6.3.4 discussed the descriptor interdependence and explained how pooling can account for it. Prior knowledge that neighbouring descriptors tend to activate similar visual words can be clearly visualised with our toy example. Let us assume that any two neighbouring descriptors located no more than 16px apart are similar as they overlap heavily. Otherwise, if located more than 16px apart, they have little or no overlap because the descriptor radius is 16px. Thus, descriptors can appear similar only if they describe repeatable image content. Figure 6.6 (b) on page 101 shows three cases of the top 1, 7, and 100 largest activations ϕ_1 per foreground bounding box responding to our first visual word (the subject's nose). Spatial deviation of the descriptor locations (also per bounding box) given 1, 7, and 100 largest ϕ_1 is indicated along the $\bar{r}$ axis. Interestingly, responses for the top 1 and 7 largest activations are induced by descriptors that are mostly up to 16px apart from each other. Allowing the top 100 largest activations reveals that descriptors inducing them are located up to 60px apart. The majority of such descriptors do not cover the subject's nose. This suggests that rejecting low value activations reduces false positives.

Improved pooling (@n). Reducing the leakage, abrupt changes in large ϕ_{kn}, and utilisation of the descriptor interdependency are addressed by simply pooling over the most significant activations given a visual word and the descriptors. This can be easily incorporated into MaxExp, ExaPro, Gamma, and AxMin pooling schemes given in equations (6.15), (6.16), (6.17), and (6.19) by using the partial sort that selects only the top @n largest values ϕ_{kn} over all $n \in \mathcal{N}$ to process, where @n is a parameter. It follows that Max-pooling is a special case, such that @$n = 1$, and a lower bound

of ExaPro that can reject the leakage. Hence, @n is a trade-off between Max-pooling (@n=1) and a chosen analytical approach, where $1 \leq @n \leq |\mathcal{N}|$. The next section shows that mid-level approaches benefit from pooling the top @n most likely activations.

Between-class separation. The overview of the pooling approaches concludes with the toy example introduced in section 6.3.5 by showing that the @n scheme increases the separation between positive and negative classes compared to other approaches. Foreground bounding boxes of faces are represented by the first atom in the dictionary. This was extracted from the subject's nose as previously outlined. Figure 6.6 (c) on page 101 presents 6 histograms of activations ϕ_1 for the first atom given three arbitrarily chosen foreground and background bounding boxes. The resulting values of Average pooling are indicated in the figure with circles and triangles corresponding to the foreground and background distributions respectively. The values of Avg@$n = 40$ are marked with crosses and diamonds. Note that Avg@$n = 40$ achieves a superior separation of foreground and background markers compared to Avg. With well adjusted @n, Avg@n (diamonds) penetrates the background distributions far to the left rejecting noise (unlike *e.g.* Max-pooling). Foreground distributions (crosses) are penetrated only marginally to the left. Thus, exploiting the shapes of these distributions improves separability. Figure 6.6 (d), also on page 101, illustrates pooling methods employed to separate the 21 foreground faces from 21 backgrounds using only pooling responses ψ_1 (one per bounding box) corresponding to the first visual word. The best separation (non-overlapping histograms) is achieved by AxMin@$n = 7$ and the worst separation by Max-pooling (histograms overlap).

6.4 Experimental Section

The coding and pooling methods are evaluated on the Caltech101 [Fei-fei et al., 2004], Flower17 [Nilsback and Zisserman, 2008b], and ImageCLEF11 [Nowak et al., 2011] datasets. Approximate Locality-constrained Soft Assignment (LcSA), Approximate Locality-constrained Linear Coding (LLC), Sparse Coding (SC), and Soft Assignment (SA) are compared. Specifically, the baseline performance of selected pooling methods is shown in section 6.4.2 and their similarity is determined using the registra-

tion from section 6.3.3. Next, the coding and pooling methods are evaluated in section 6.4.3. LcSA, LLC, and SC mid-level features are processed by Max-pooling (Max), Gamma Correction (Gamma), *theoretical expectation of Max-pooling* (MaxExp), its approximation AxMin, and *at least one visual word* $\boldsymbol{m}_k$ *being present in image* i (ExaPro). Mix-order Max-pooling (MixOrd) and the ℓ_p norm (lp-norm) are also briefly investigated. The @n scheme from section 6.3.5 is applied to AxMin, ExaPro, and MaxExp to demonstrate it can improve classification performance. The impact of the dictionary size and performance of the coding optimisations from section 6.2.7 are also measured.

6.4.1 Experimental Arrangements and Datasets

The Caltech101 set [Fei-fei et al., 2004] consists of 101 classes of objects which are aligned to the centres of images, as well as a separate background class. The majority of evaluations are performed with 15 training images per class (unless otherwise stated).

The Flower17 set [Nilsback and Zisserman, 2008b] of 17 flower classes was used for further evaluations (data splits are supplied for this corpus).

The ImageCLEF11 Photo Annotation set [Nowak et al., 2011] is a challenging collection of images represented by 99 concepts of a varied nature, including complex topics, *e.g. party life*, *funny*, *work*, *birthday*. Unlike sets of objects, this challenge aims at annotation labels that correspond to human-like understanding of a scene. ImageCLEF11 is a subset of MIRFLICKR with vastly improved annotations which enables better classification [Huiskes and Lew, 2008, Huiskes et al., 2010]. To evaluate the mid-level coding and pooling methods in a simple framework, only Opponent SIFT on a dense grid was used for this set. Only the visual annotation was used in this study. Moreover, the training set was doubled by left-right flipping training images [Chatfield et al., 2011].

The PascalVOC07 set [Everingham et al., 2007] consists of 20 classes of objects of varied nature, *e.g. human*, *cat*, *chair*, *train*, *bottle*. This is a challenging collection of images with objects that appear at variable scales and orientations, often in difficult visual contexts and backgrounds, being frequently partially occluded. The training, validation, and testing splits as provided for this corpus.

Dataset	Splits no.	Train+Val. samples	Test samples	Total images	Dict. size	Descr. type/ dimensions
Caltech101	10x	12+3=15/24+6=30	rest	9144	4K	SIFT/128D
Flowers17	3x	680+340=1020	340	1680	4K	Opp. SIFT/ 192D
ImageCLEF11	1x	6K+2K=8K	10K	18K	16K	
PascalVOC07	1x	2501+2510=5011	4952	9963	4K-40K	SIFT/128D

	Descr. interval	Radii (px)	Descr. per img.	Spatial/other schemes	Kernel types	Classifier used
Caltech101	4,6,8,10px	16,24, 32,40	5200	none/SCC/SPM	linear	multiclass
Flowers17	8,14,20,26		7900	SCC		
ImageCLEF11	8,12,16,20		4400	SCC/SPM/ DoPM	linear/ χ^2_{RBF}	multilabel
PascalVOC07	4,6,8,10, 12,14,16	12,16,24,32, 40,48,56	19420		linear	

Table 6.2: Summary of the datasets, descriptor parameters, and experimental details.

To summarise the experimental arrangements, a variety of parameters are collected in the given above table 6.2.

Dictionary. K-means was used throughout the experiments. However, Fast Hierarchical Nearest Neighbour Search, described in section 6.2.7, employs 64×64 and 128×128 hierarchical k-means on Caltech101 and ImageCLEF11. Moreover, Online Dictionary Learning for SC [Mairal et al., 2010] is used to train dictionaries for PascalVOC07.

Dataset bias. Spatial relations in images were exploited by either Spatial Coordinate Coding (SCC) described in chapter 5 or Spatial Pyramid Matching (SPM) from [Lazebnik et al., 2006]. SPM used 4 levels of coarseness with 1×1, 2×2, 3×3, and 4×4 grids on Caltech101 and IamgeCLEF11. Also, SPM was set to 3 levels of coarseness with 1×1, 1×3, 3×1, and 2×2 grids for PascalVOC07. Dominant Angle Pyramid Matching (DoPM) from chapter 5 was used to exploit dominant edge bias in ImageCLEF11 and PascalVOC07. DoPM used 5 levels of coarseness with 1, 3, 6, 9, and 12 grids. Moreover, DoPM employed SCC by default for PascalVOC07.

Kernels. Linear kernels $Ker_{ij} = (\boldsymbol{h}_i)^T \cdot \boldsymbol{h}_j$ were used, where $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^{KQ}$ are image signatures for $i, j \in \mathcal{I}$. The χ^2 distance merged with the RBF kernel (χ^2_{RBF}) defined as $Ker_{ij} = exp\left[-\rho^2 \sum_k (h_{ki} - h_{kj})^2/(h_{ki} + h_{kj})\right]$ was also used, $1/\rho$ is the RBF radius.

Classifier. Multi-class KDA [Tahir et al., 2009] was applied to both Clatech101 and Flower17 to process kernels formed from different mid-level feature and pooling variants.

Mean Accuracy is the reported performance measure. Multi-label KDA [Tahir et al., 2009] was applied to ImageCLEF11 and PascalVOC07, as it was previously found to be a robust performer on these sets [Tahir et al., 2010]. Due to the multi-label nature of ImageCLEF11, Mean Average Precision (MAP) is used to report the performance.

6.4.2 Baseline Performance and Registration between Gamma/AxMin and MaxExp.

The baseline performance of LcSA mid-level coding paired with various pooling methods is determined for Caltech101 (15 training images/class, no spatial information). Several sets of image signatures are computed on the training data for Gamma, AxMin, and MaxExp pooling given several values of their parameters γ, β, and $\bar{N}$. Next, registration between the signatures of Gamma/AxMin and MaxExp is performed by minimising equation (6.18) from section 6.3.3. For each $\bar{N}$, a corresponding γ and β is found. Figure 6.7 (a) shows the classification results on both validation and test sets. Results for MaxExp, Gamma, and AxMin pooling are shown as functions of the

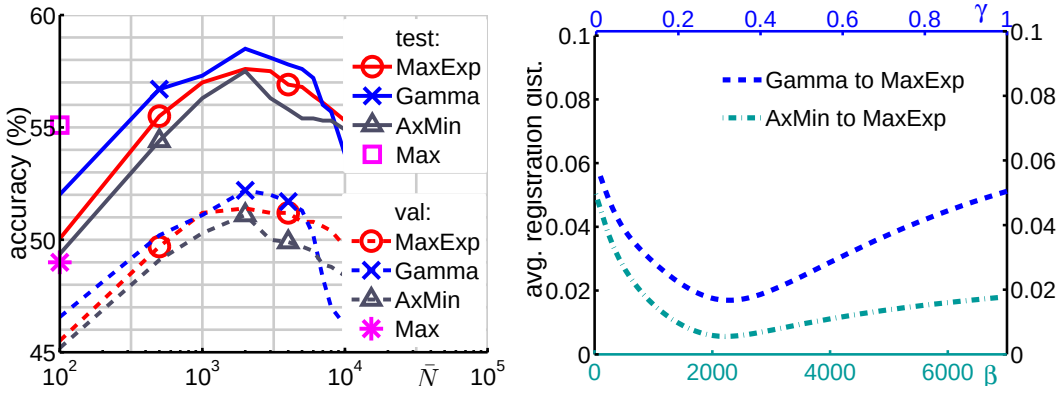


Figure 6.7: Baseline LcSA mid-level coding (Caltech101, 15 training images/class, no spatial information, linear kernels). (*a*) LcSA with Max, MaxExp, Gamma, and AxMin pooling. Gamma and AxMin are brought to the MaxExp parameter space $\bar{N}$ by registration using equation (6.18) from section 6.3.3. Dashed and solid curves show the validation and test results. (*b*) Corresponding low average registration distance between Gamma/AxMin and MaxExp signatures highlights their closeness.

common parameter $\bar{N}$ due to the registration. The three curves shown have peak performance for the same value of $\bar{N}$, indicating that Gamma and AxMin act on mid-level features similar to MaxExp. This supports our discussion in sections 6.3.3 and 6.3.4 regarding the common theoretical basis of these methods. Figure 6.7 (b) shows the average Euclidian registration distance between Gamma/AxMin and MaxExp signatures as a function of parameters γ and β. Parameters $\gamma = 0.32$ and $\beta = 2200$ indicate the attained minima and correspond to the optimal $\bar{N} = 2000$ selected from plot 6.7 (a).

Further, figure 6.7 (a) shows the baseline Max-pooling accuracy of 55.1% on the test set. Gamma improved on this score by 3.4%, reaching 58.5% accuracy. The Average pooling is not reported in the following sections as it scored only 42.6% accuracy and consistently underperformed. Note that peaks in accuracy on the validation and test sets match each other closely. Thus, only performance achieved on test sets is reported in further sections. However, various parameters of the classification pipeline were determined during cross-validation on validation sets.

Lastly, figure 6.8 shows the classification results for the baseline Max-pooling as a function of LcSA coding parameters σ and l, respectively. Caltech101 (15 images/class, Spatial Pyramid Matching) and ImageCLEF11 (Spatial Coordinate Coding) were evaluated both on linear kernels. The best coding parameters, indicated by crosses, seem to correlate well with the minima of ξ^2, as indicated by diamonds. The above parameters were found by evaluating equation (6.10) given $156K$ descriptors per dataset that were drawn at random.

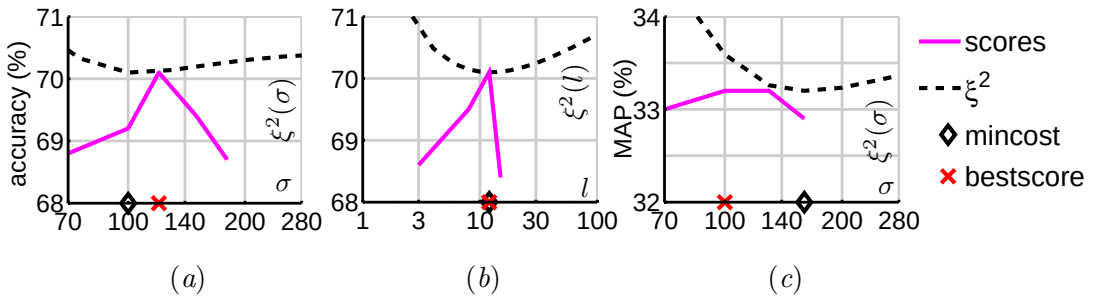
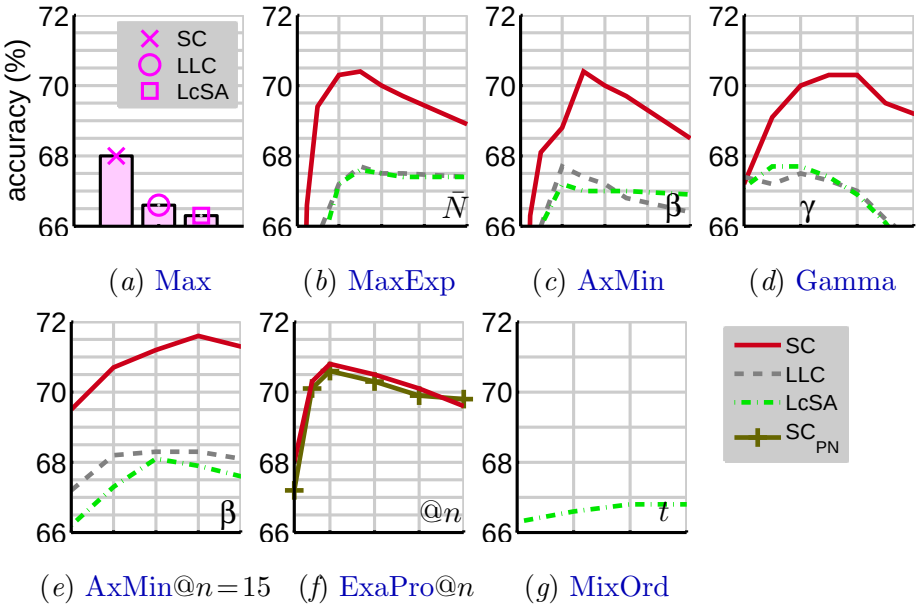


Figure 6.8: ξ^2 quantisation loss (dashed curves) and classification results (solid curves) as functions of σ and l. We varied (*a*) σ, (*b*) l on Caltech101, (*c*) σ on ImageCLEF11. Diamonds and crosses indicate the minima of ξ^2 and the best results.



(*a*) Max (*b*) MaxExp (*c*) AxMin (*d*) Gamma

(*e*) AxMin@$n=15$ (*f*) ExaPro@n (*g*) MixOrd

Figure 6.9: Performance of mid-level coding methods LcSA, LLC, and SC given pooling methods (Caltech101, 15 images/class, Spatial Coordinate Coding, linear kernels). The following are (*a*) baseline Max-pooling, (*b*) MaxExp pooling as a function of $\bar{N}$, (*c*) its close approximation AxMin pooling w.r.t. β, (*d*) Gamma pooling given γ, (*e*) AxMin@$n=15$ as a function of β, (*f*) ExaPro@n for positive (in solid) and positive-negative activations (SC_{PN}) of SC as discussed in section 6.2.3, and (g) MixOrd pooling.

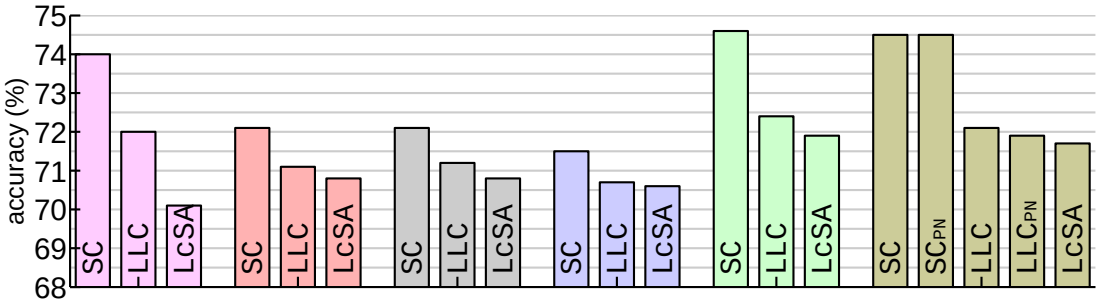
6.4.3 Evaluations of Mid-level Coding and Pooling Methods

We describe now how the coding and pooling methods performed in a practical classification scenario. The impact of pooling parameters on the results is shown first. They are indicated in plots, *e.g.* $\bar{N}$, β, @n. Next, the best scores of each coding and pooling pair are reported to facilitate comparisons, *e.g.* @$n=3$, 5, or 7 means @n is fixed in a given experiment. Additional components and kernel choices are also indicated.

Caltech101. Figure 6.9 introduces results for the coding and pooling methods as functions of the pooling parameters (15 training images/class, Spatial Coordinate Coding). Note that there are no erratic variations in plots. The best performance for each method corresponds to the peak of each curve (peaks on the validation and test sets also matched each other). Plot 6.9 (a) shows that the baseline Max-pooling yields $68.0 \pm 0.5\%$, $66.6 \pm 0.4\%$, and $66.3 \pm 0.3\%$ accuracy for SC, LLC, and LcSA, respec-

tively. Plots 6.9 (b-d) show the accuracy for MaxExp, AxMin, and Gamma. SC yields 70.4±0.4% accuracy for all three schemes. LLC and LcSA achieve 67.7±0.5% accuracy with AxMin and Gamma, respectively. Improvements over Max-pooling given SC, LLC, and LcSA amount to 2.4%, 1.1%, and 1.4%, respectively. Note that MaxExp scored best for $\bar{N} \approx 3000 < 5200$ (mean descriptor count). Figure 6.9 (e) shows that AxMin@$n=15$ with SC yields 71.6±0.4% giving a 3.4% improvement over Max-pooling due to the @n scheme. LLC and LcSA score 68.3±0.4% and 68.1±0.5%. Figure 6.9 (f) shows scores for ExaPro@n and SC that amount to 70.8±0.3% and 70.6±0.3% given the positive and positive-negative activations respectively. As suggested in section 6.2.3, no benefits of allowing $\phi_k < 0$ were observed. Next, plot (g) shows MixOrd given LcSA ($t = 1, 3, 5, 7$ signatures per image were combined as described in section 6.3.1). This resulted in an 0.8% increase over Max-pooling. Not included in the plots, lp-norm and LcSA yields 66.4±0.5% at best, ExaPro and LLC yields 68.2±0.5%.

Figure 6.10 shows additional performance results of coding and pooling (15 training images/class, Spatial Pyramid Matching). Plot 6.10 (a) shows that the baseline Max-pooling scores 74.0±0.3%, 72.0±0.5% and 70.1±0.4% given SC, LLC, and LcSA. Plots 6.10 (b-d) show scores for MaxExp, AxMin, and Gamma. Performance of SC and LLC deteriorated for these three schemes. LcSA scores 70.8±0.5%, yielding a



(*a*) Max (*b*) MaxExp (*c*) AxMin (*d*) Gamma (*e*)AxMin@$n=3$ (*f*) ExaPro@n

Figure 6.10: Performance of mid-level coding methods LcSA, LLC, and SC given pooling methods (Caltech101, 15 images/class, Spatial Pyramid Matching, linear kernels). SC, LLC, and LcSA are paired with (*a*) baseline Max-pooling, (*b*) MaxExp, (*c*) AxMin, (*d*) Gamma, (*e*) AxMin@n=3, and (*f*) ExaPro@n. SC_{PN} and LLC_{PN} show results for SC and LLC given the positive-negative activations.

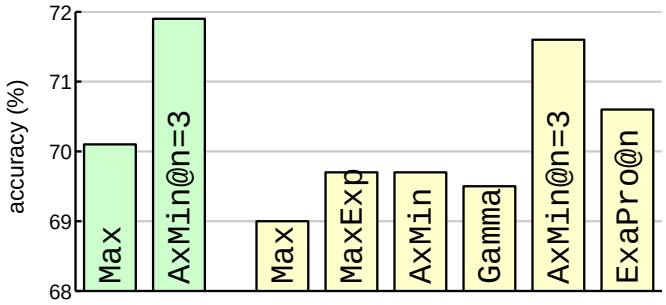
small improvement. Plot 6.10 (e) shows the positive impact of AxMin@n=3 on the coding methods. SC and LLC improve marginally from 74.0±0.3% and 72.0±0.5% given Max-pooling to 74.6±0.4% and 72.4±0.5% accuracy. LcSA yields 71.9±0.4% giving a 1.8% improvement over Max-pooling. Plot 6.10 (f) shows ExaPro@n with SC reaching 74.5±0.4% and LLC achieving 72.1±0.3%. Note that allowing positive-negative activations does not improve the performance. Not in the plots, lp-norm and MixOrd yield 70.3±0.3% and 70.1±0.4% at best. Table 6.3 summarises the best scores achieved by this study on Caltech101 (15 and 30 training images/class). See appendix A.4 for a statistical significance test. Our results can be compared to various results achieved by others in table 6.4. The best results reported in the literature are Group-Sensitive Multiple Kernel Learning (GS-MKL) [Yang et al., 2012a] with performance of 84.3%, Discriminative Affine Sparse Codes (ASIFT) [Kulkarni and Li, 2011] with 83.3%, Multi-way SVM on appearance and shape features (M-SVM) [Bosch et al., 2007] with 81.3%, and Graph-matching Kernel (GMK) [Duchenne et al., 2011] with 80.3% accuracy.

	SA AxMin@n	LcSA AxMin@n	LcSA Max
SCC (15)	67.8±0.6	68.1±0.5	66.3±0.3
SPM (15)	71.6±0.4	71.9±0.4	70.1±0.4
SPM (30)	78.6±0.5	78.8±0.4	77.8±0.3
	LLC AxMin@n	SC AxMin@n	SC Max
SCC (15)	68.3±0.4	**71.6±0.4**	68.0±0.5
SPM (15)	72.4±0.5	**74.6±0.4**	74.0±0.3
SPM (30)	79.5±0.5	**81.3±0.6**	80.4±0.6

Table 6.3: Summary of our best results on Caltech101. The first column indicates how the spatial information was injected. Numbers of training images per class are indicated in brackets.

[Boureau et al., 2010b]	HA, 1K, MaxExp	71.8±0.8
[Chatfield et al., 2011]	HA, 8K, Avg+χ^2	74.2±0.6
[Chatfield et al., 2011]	SA, 8K, Avg+χ^2	75.9±0.6
[Lingqiao et al., 2011]	LcSA, 1K, Max	76.5±0.7
[Yang et al., 2009]	LLC, 1K, Max	73.4
[Chatfield et al., 2011]	LLC, 8K, Max	76.9±0.4
[Yang et al., 2009]	SC, 1K, Max	73.2±0.5
[Boureau et al., 2010b]	SC, 1K, Max, MF	75.1±0.9
[Boureau et al., 2011]	SC, 1K x64, CSP	77.1±0.7
[Chatfield et al., 2011]	Fisher, 256x256, Gamma	77.8±0.6
[Duchenne et al., 2011]	GMK	80.3±1.2
[Bosch et al., 2007]	M-SVM	81.3±0.8
[Kulkarni and Li, 2011]	ASIFT	83.3
[Yang et al., 2012a]	GS-MKL	84.3

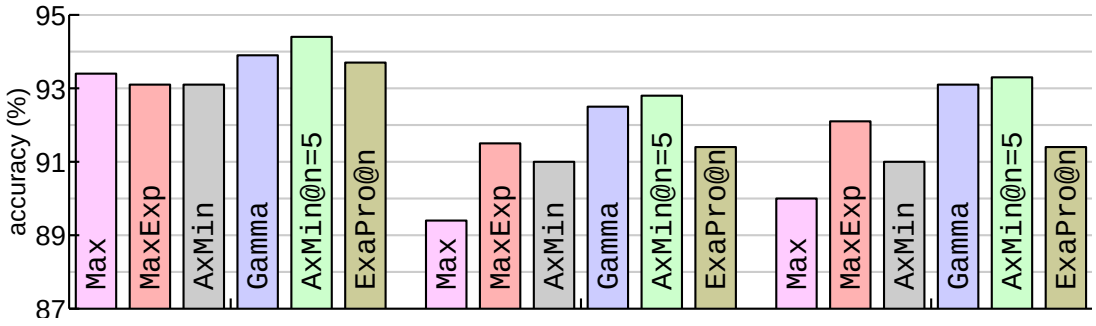
Table 6.4: Results on Caltech101 (30 training images/class) reported in the literature. Mid-column: coding type, signature length, and pooling. MF are Macrofeatures [Boureau et al., 2010b], CSP is Pooling in Configuration Space [Boureau et al., 2011]. The last four rows show the highest results (acronyms are explained in the text).



(*a*) SA (*b*) LcSA

Figure 6.11: SA scores low given Max-pooling, MaxExp, AxMin, and Gamma. Note, SA and LcSA perform similar for AxMin@n=3.

Soft Assignment and Leakage. Section 6.3.5 discussed Soft Assignment and the problem of the inherent leakage in this method. The experimental findings are shown in figure 6.11 (Caltech101, 15 training images/class, Spatial Pyramid Matching) and present SA given a variety of pooling methods. SA scores only 69.0±0.6% accuracy given Max-pooling. MaxExp, AxMin, and Gamma yield small improvements. However, applying AxMin@n=3 to SA yields a 2.6% improvement over Max-pooling leading to 71.6±0.4% accuracy. For comparison, LcSA with AxMin@$n=3$ scores 71.9±0.4%. Note that Max-pooling scores poorly despite being a special case of @n pooling, *e.g.* AxMin@n=1. We suspect that exploiting the descriptor interdependency (@$n>1$), as outlined in section 6.3.5, is important in tackling the leakage.



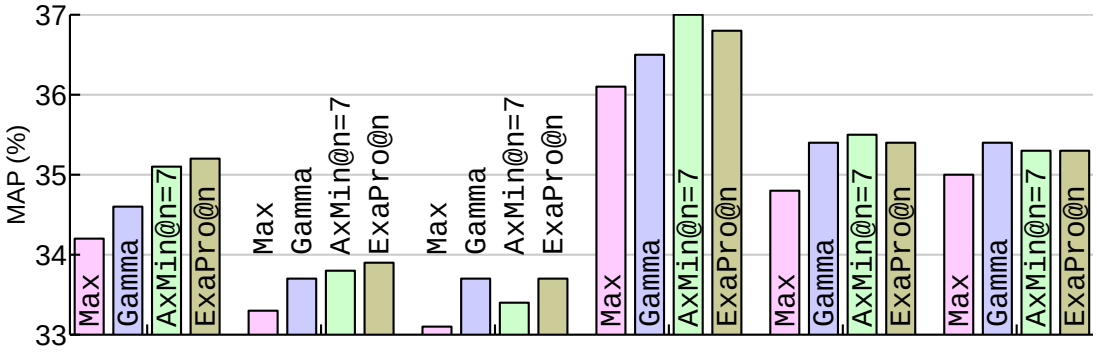
(*a*) SC (*b*) LLC (*c*) LcSA

Figure 6.12: Performance of mid-level coding methods for various pooling schemes (Flower17, Spatial Coordinate Coding, linear kernels). Plots (*a-c*) show results for SC, LLC, and LcSA. Note that the majority of pooling schemes outperform Max-pooling.

Flower17. Plots 6.12 (a-c) show results for SC, LLC, and LcSA for various pooling schemes (Spatial Coordinate Coding, linear kernels). Plot 6.12 (a) shows that SC combined with either MaxExp or AxMin has a performance below the baseline Max-pooling which yields 93.4±0.3%. However, SC with Gamma gives 93.9±1.6% accuracy. SC with AxMin@n=5 scores 94.4±0.4%. LLC in plot 6.12 (b) also improves over its baseline of 89.4±1.6% accuracy reaching 92.6±1.8% and 92.8±0.5% for Gamma and AxMin@n=5, which is a 3.4% improvement. LcSA in plot 6.12 (c) scores 93.1±1.1% and 93.3±0.5% accuracy for Gamma and AxMin@n=5. This is a 3.3% improvement over the Max-pooling baseline of 90.0±0.2%. Table 6.5 summarises our results. See appendix A.4 for a statistical significance test. The results from chapter 5 are 91.4% accuracy. The other studies are [Nilsback and Zisserman, 2008b] with 88.3%, [Liu et al., 2011] with 88.2%, and [Yan et al., 2010] with 86.7% accuracy.

	LcSA	LLC	SC
Max	90.0±0.2	89.4±1.6	**93.4±0.3**
Gamma	93.1±1.1	92.5±1.1	**93.9±1.6**
AxMin@n	93.3±0.5	92.8±0.8	**94.4±0.4**

Table 6.5: The best results attained by us on Flower17 (Spatial Coordinate Coding and linear kernels were used). Max, Gamma, and AxMin@n=5 pooling are evaluated.



(*a*) SC_{LIN} (*b*) LLC_{LIN} (*c*) $LcSA_{LIN}$ (*d*)$SC\chi^2_{RBF}$ (*e*)$LLC\chi^2_{RBF}$ (*f*)$LcSA\chi^2_{RBF}$

Figure 6.13: Performance of mid-level coding and pooling (ImageCLEF11, Spatial Coordinate Coding). SC, LLC, and LcSA are paired with Max-pooling, Gamma, AxMin@n=7, and ExaPro@n. We used (*a-c*) linear and (*d-f*) χ^2_{RBF} kernels.

ImageCLEF11. To conclude the coding and pooling experiments on a challenging set, SC, LLC, and LcSA are paired with Max-pooling, Gamma, AxMin@$n=7$, and ExaPro@n. MaxExp and AxMin are not reported as they perform similar to Gamma. Spatial Coordinate Coding was used in these tests. Plots 6.13 (a-c) show results on linear kernels. Max-pooling scores 34.2%, 33.3%, and 33.0% MAP given SC, LLC, and LcSA. Figure 6.13 (a) shows AxMin@n=7 and ExaPro@n yield 35.1% and 35.2% MAP for SC. This gives a 1% improvement over Max-pooling (the best result on linear kernels). LLC and LcSA yield 33.9% and 33.8% MAP for ExaPro@n and Gamma.

Plots 6.13 (d-f) show results on χ^2_{RBF} kernels that improve performance further. Plots 6.13 (b) show that Max-pooling yields 36.1%, 34.9%, and 35.0% MAP given SC, LLC, and LcSA. Next, AxMin@n=7 scores 37.0% (the best result on χ^2_{RBF} kernels). This is 0.9% improvement over Max-pooling. Lastly, LLC and LcSA yield 35.5% and 35.4% MAP given AxMin@n=7 and Gamma. The evaluated pooling schemes improved results over the baseline on both kernel types. We note a trend that LcSA works well with Gamma (also MaxExp and AxMin in previous sections). SC and LLC tend to benefit more from AxMin@n and ExaPro@n. Also, LLC and LcSA yield very similar results.

ImageCLEF11 and Bias in Images. Given the complexity of ImageCLEF11, Spatial Pyramid Matching (SPM) and Dominant Angle Pyramid Matching (DoPM) were employed for the final experiments (Sparse Coding, AxMin@$n=7$, linear and χ^2_{RBF} kernels used). Table 6.6 shows results for SPM and DoPM. Given linear kernels, they have a performance of 35.2% and 35.3% MAP. For χ^2_{RBF}, they yield 36.7% and 36.8% MAP. Furthermore, combining either SCC (scored 37.0%) or SPM with DoPM yields 38.4% MAP. Only Opponent SIFT on a dense grid is used. The best results in previous studies for the visual configuration are 38.8% [Binder et al., 2011] (multiple interest points, descriptors, and kernels combined) and 38.2% [Su and Jurie, 2011] (multiple semantic contexts, SPM channels, semantic features, and kernels combined).

	SCC	SPM	DoPM	Comb.
linear	35.1	35.2	35.3	36.6
χ^2_{RBF}	**37.0**	36.7	36.8	**38.4**

Table 6.6: Our best results on ImageCLEF11 (Sparse Coding and AxMin@n=7). First column: kernel type. First row: bias type. Comb. denotes SPM and DoPM combined.

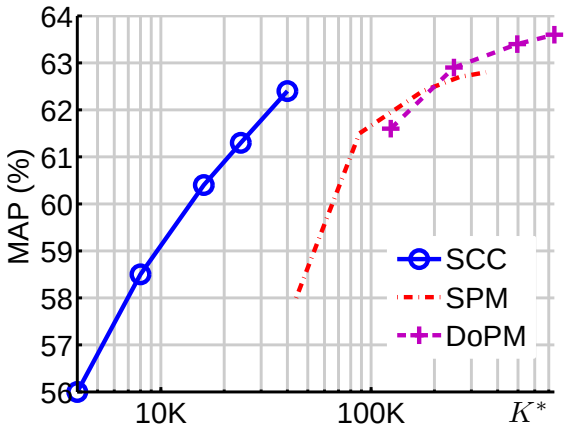
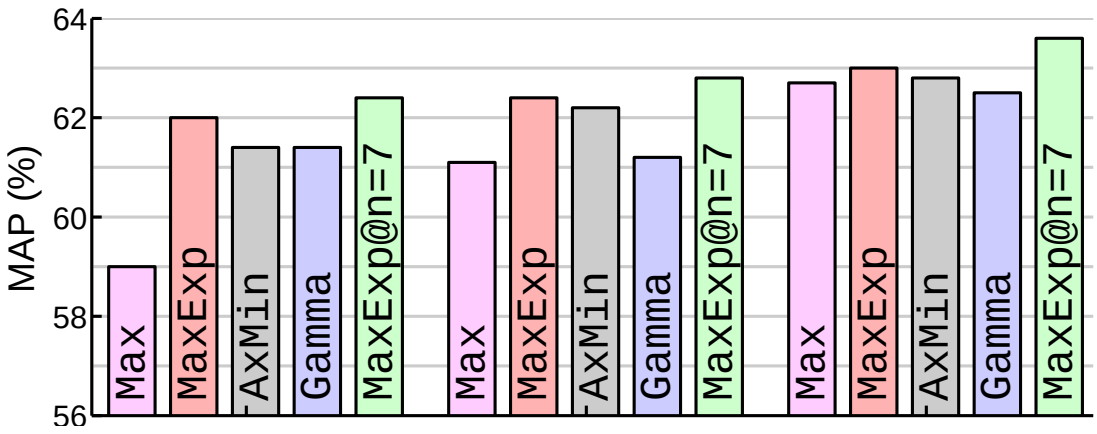


Figure 6.14: Evaluation of SCC, SPM, and DoPM approaches on the PascalVOC07 set. The overall signature length K^* is indicated. Linear kernels and MaxExp@n=7 are used for this experiment.

PascalVOC07 and Bias in Images. Figure 6.14 compares the classification performance of SCC, SPM, and DoPM approaches on the PascalVOC07 set given various dictionary sizes. Linear kernels and MaxExp@n=7 are used for this experiment. The dictionary size is varied from $4K$ to $40K$ atoms for SCC. The signature lengths are the same as the dictionary sizes in this case. The highest result attained by SCC amounts



(a) SCC (b) SPM (c) DoPM

Figure 6.15: Evaluation of SCC, SPM, and DoPM schemes on the PascalVOC07 set given Max-pooling, MaxExp, AxMin, Gamma, and MaxExp@$n=7$. The dictionary sizes are $40K$, $32K$, and $24K$ atoms for SCC, SPM, and DoPM.

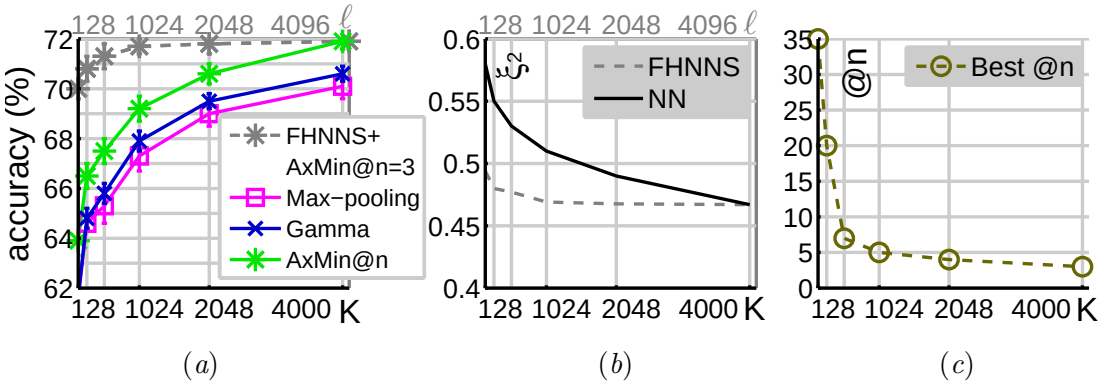
to 62.4% MAP. Moreover, we vary the dictionary size from $4K$ to $32K$ atoms for SPM. This results in the signature lengths between $44K$ and $352K$. The best result attained by SPM amounts to 62.8% MAP. Lastly, the dictionary size is varied from $4K$ to $24K$ atoms for DoPM. The corresponding signature lengths are between $124K$ and $744K$. This method scores 63.6% MAP.

Figure 6.15 demonstrates various pooling strategies given dictionary sizes of $40K$, $32K$, and $24K$ atoms for SCC, SPM, and DoPM approaches, respectively. Firstly, we discuss SCC approach. MaxExp@n=7 scores 62.4% MAP followed closely by MaxExp that yields 62.0% MAP. AxMin and Gamma attain the same score of 61.4% MAP followed by Max-pooling that yields 59.0% MAP only. Next, we discuss SPM approach. MaxExp@n=7 scores 62.8% MAP followed closely by MaxExp and AxMin that yield 62.4% and 62.2% MAP. Gamma and Max-pooling attain 61.2% and 61.1% MAP only. Lastly, we discuss DoPM approach. MaxExp@n=7 scores 63.6% MAP followed by MaxExp and AxMin that yield 63.0% and 62.8% MAP. Max-pooling attains 62.7% MAP and outperforms Gamma that yields 62.5% MAP only.

To conclude, the SCC approach results in very competitive signature lengths. However, the coding step is computationally prohibitive for large visual dictionaries. It takes 815 and 3.6 seconds to code 1000 descriptors on a single 2.3GHz AMD Opteron core given $40K$ and $4K$ atoms, respectively. This may be partially addressed by the FHNNS scheme proposed earlier. SPM achieves a marginally better performance with somewhat smaller dictionaries at a price of larger image signatures. DoPM achieves the best performance at a price of sizeable image signatures. Furthermore, we observe that the @n scheme combined with MaxExp attains the highest scores amongst the investigated pooling strategies. MaxExp and its approximation AxMin are also strong performers followed by Gamma and Max-pooling. These results remain consistent.

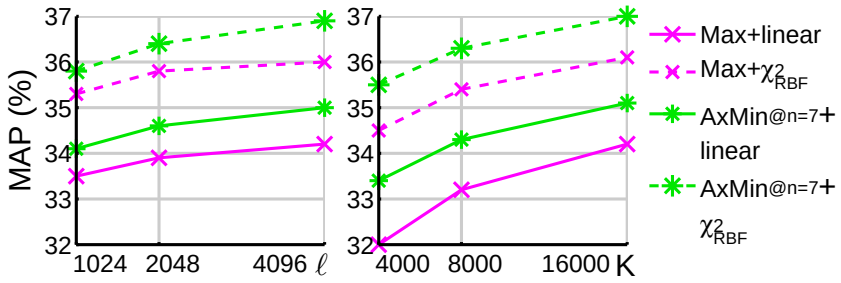
Dictionary Size and Fast Hierarchical Nearest Neighbour Search.

To conclude these evaluations, there follows a brief investigation into: i) the impact of the dictionary size on LcSA and SC, ii) Fast Hierarchical Nearest Neighbour Search (FHNNS), outlined in section 6.2.7, paired with LcSA and SC.



(a) (b) (c)

Figure 6.16: Performance of LcSA given Fast Hierarchical Nearest Neighbour Search (section 6.2.7) and ordinary NN (Caltech101, 15 training images/class, Spatial Pyramid Matching). (*a*) LcSA with FHNNS as a function of ℓ (cluster dilation). Also, LcSA with NN as a function of K (dictionary size) for Max, Gamma, and AxMin@n. (*b*) Corresponding quantisation errors ξ^2. (*c*) The optimal value @n for AxMin@n as a function of the dictionary size K.



(*a*) SC+FHNNS (*b*) SC

Figure 6.17: Performance of SC given FHNNS and ordinary NN (ImageCLEF11, Spatial Coordinate Coding). We applied linear and χ^2_{RBF} kernels to Max-pooling and AxMin@n=7 based signatures. (*a*) SC with FHNNS as a function of ℓ (cluster dilation). (*b*) SC with NN as a function of K (dictionary size).

Dictionary Size. Figure 6.16 (a) shows the performance on Caltech101 (15 training images/class, Spatial Pyramid Matching, linear kernels used) for LcSA given Max-pooling, Gamma, and AxMin@n. The dictionary size K was varied. Max-pooling and Gamma perform similar for $K \in \langle 128; 512 \rangle$. Gamma scores marginally better than Max-pooling for larger K. AxMin@n appears a strong performer even for small K. Plot 6.16

(c) shows how the best performing parameter @n of AxMin@n varies as a function of K. Figure 6.17 (b) shows that ImageCLEF11 (SC, Spatial Coordinate Coding, χ^2_{RBF} kernels used) benefits from a larger dictionary.

FHNNS. Figure 6.16 (a) also presents the results for LcSA with FHNNS and AxMin@n= 3 using $K' = 4096$ atoms. Given $\ell \ll K'$ (ℓ impacts the cluster dilation), LcSA and FHNNS had a higher performance than LcSA and Nearest Neighbour. The first approach searches through only ℓ anchors to code a descriptor. However, it still produces K' long mid-level features. The latter method searches through $K = \ell$ anchors and produces only K long features in a comparable coding time. Hence, its performance drops for small values of K. Plot 6.16 (b) shows the corresponding quantisation error for LcSA with FHNNS is smaller when compared to LcSA with NN (assuming $K = \ell \ll K'$). Lastly, figure 6.17 (a) presents the classification results for SC with FHNNS on ImageCLEF11. Given $\ell = 4096$ and $K' = 16384$, this method is as robust as ordinary SC in figure 6.17 (b) and saves on computational cost (see table 6.1).

6.4.4 Discussion on the Coding and Pooling Approaches

Mid-level coding methods differ both in their classification performance (section 6.4.3) and computational cost (table 6.1). SA, LcSA, LLC, and SC exhibited varied performance depending on the pooling variant. Further, a strong relation is observed between Gamma and MaxExp pooling, as discussed in section 6.3.3, and empirically validated in figures 6.7 (a, b). Classification experiments also suggest these two methods are similar. In practice, using a carefully selected pooling methods led to significant improvements over the baseline Max-pooling approach. Specifically, LcSA and LLC benefited from MaxExp, AxMin, Gamma, and the @n pooling schemes. SC and SA demonstrated their best performance during the classification when paired with the @n scheme. This may be attributed to the leakage suppression discussed in section 6.3.5. Furthermore, carefully selected pooling parameters led to the best classification performance by accounting for the descriptor interdependence, as outlined in sections 6.3.4 and 6.3.5. AxMin@n, MaxExp@n, and ExaPro@n are examples of extending AxMin, MaxExp, and ExaPro pooling with the @n scheme. Note that SC consistently outperformed

LcSA and LLC, but at the price of higher computational cost. Regarding computational efficiency, FHNNS, from section 6.2.7, benefited the coding as shown in section 6.4.3. Combining LcSA and SC with FHNNS improved their computational speed 4× and 1.5× (table 6.1) with no observable decline in the classification results. Large overlap between the k-means dictionary clusters was required to limit the quantisation noise along the cluster boundaries. Lastly, the impact of Spatial Coordinate Coding, Spatial Pyramid Matching, and Dominant Angle Pyramid Matching on the classification quality was evaluated. Due to the compactness of mid-level features generated with SCC, it thrived on the discriminative properties of the @n scheme, as explained in section 6.3.5. Note that computing kernels from SCC based signatures, as proposed in chapter 5, was 36× faster than using SPM signatures. Moreover, SCC yielded better performance than SPM on ImageCLEF11. Combining SCC/SPM and DoPM gave the best final performance.

Pipeline Variants. For rapid classification, LcSA or LLC with FHNNS, MaxExp or Gamma pooling, Spatial Coordinate Coding, and a linear kernel is effective. For large complex datasets, SC, AxMin@n or MaxExp@n, SPM, DoPM, and χ^2_{RBF} kernels may be used. For small datasets, SC, AxMin@n or MaxExp@n, SCC, and a linear kernel are a good choice.

6.5 Conclusions

This chapter presented an extensive comparison of four widely used mid-level coding schemes on three popular datasets. Various pooling strategies were evaluated to asses their impact on classification. We demonstrated that the performance of SA, LcSA, LLC, and SC schemes depends on the choice of pooling. Evaluated MaxExp, Gamma, AxMin, and ExaPro improved the performance over the baseline Max-pooling scheme. Furthermore, we proposed a simple extension termed @n which is applicable to these pooling schemes. Its positive impact on performance with AxMin@n, MaxExp@n, and ExaPro@n pooling is observed. SC outperformed SA, LcSA, and LLC on the evaluated datasets leading to 81.3% accuracy on Caltech101, 94.4% accuracy on Flower17, 38.4% MAP on ImageCLEF11 (visual configuration, Opponent SIFT used only), and 63.6%

MAP on PascalVOC07. LLC and LcSA were close competitors. Possible extensions of this work include combining the proposed pooling schemes with Fisher Vector Encoding. An optimisation of the pooling parameters on the classifier level is also possible.

Chapter 7

Visual Categorisation Beyond First-order Occurrence Pooling on Mid-level Features.

In object recognition, the ubiquitously popular Bag-of-Words model assumes: i) extraction of local descriptors from images, ii) embedding these descriptors by a coder to a given visual vocabulary space which results in so-called mid-level features, iii) extracting statistics from mid-level features with a pooling operator that aggregates occurrences of visual words in images into signatures suitable for classification. As the last step aggregates only occurrences of visual words represented by coefficients of each mid-level feature vector, we refer to it as First-order Occurrence Pooling. However, this chapter proposes to aggregate over co-occurrences of visual words in mid-level features. This is termed as Second-order Occurrence Pooling. Moreover, we provide a derivation of Second- and Higher-order Occurrence Pooling based on linearisation of so-called Minor Polynomial Kernel and generalise it to work with a variety of pooling operators: Average, Max-pooling, Analytical pooling, and a highly effective trade-off between Max-pooling and Analytical pooling. We evaluate how First-, Second-, and Third-order Occurrence Pooling performs given various coders and pooling operators. For bi- and multi-modal coding with two or more coders, we propose an extension of Second- and Higher-order Occurrence Pooling based on linearisation of Minor Polyno-

mial Kernel. We demonstrate, by combining both the grey scale and colour mid-level features, that such a linearisation outperforms naive fusing schemes. We illustrate that the well-known Spatial Pyramid Matching in Bag-of-Words and other similar methods are special cases of this method. Lastly, we compare the proposed approaches to other renowned methods (*e.g.* Fisher Vector Encoding) in the same testbed and attain state-of-the-art results.

7.1 Introduction

Bag-of-Words proposed in [Sivic and Zisserman, 2003, Csurka et al., 2004] is a popular approach which transforms local image descriptors [Lowe, 1999, Mikolajczyk and Schmid, 2005, van de Sande et al., 2008] into image representations that are used in matching and classification. Its first implementations were associated with object retrieval and scene matching [Sivic and Zisserman, 2003], as well as visual categorisation [Csurka et al., 2004]. The BoW approach has undergone tremendous changes over recent years. To date, a number of its variants have been developed and reported to produce state-of-the-art results: Kernel Codebook [van Gemert et al., 2008, 2010, Philbin et al., 2008] a.k.a. Soft Assignment and Visual Word Uncertainty, Approximate Locality-constrained Soft Assignment proposed in chapter 6 as well as in [Lingqiao et al., 2011], Sparse Coding [Lee et al., 2007, Yang et al., 2009], Linear Coordinate Coding [Yu et al., 2009], Approximate Locality-constrained Linear Coding [Wang et al., 2010], Laplacian Sparse Coding [Gao et al., 2010], and Over-Complete Sparse Coding [Yang et al., 2010]. We refer to this group of BoW as the first group. Recently, Super Vector Coding [Zhou et al., 2010], Vector of Locally Aggregated Descriptors [Jégou et al., 2010], Fisher Vector Encoding (FK) proposed in [Perronnin and Dance, 2007, Perronnin et al., 2010], and Vector of Locally Aggregated Tensors (VLAT) from [Negrel et al., 2012] have emerged as challenging competitors compared to *e.g.* Sparse Coding [Yang et al., 2009]. For distinction, we call them the second group. The main hallmarks of these approaches are: i) their coding step encoding descriptors with respect to the cluster centres after assigning them to these clusters, ii) second-order statistics (last two methods) extracted from mid-level features in order to complement the first-order cues, iii) their pooling

step benefiting from Power Normalisation (PN) used by [Boughorbel et al., 2005, Perronnin et al., 2010, Jégou et al., 2009], also introduced as Gamma in chapter 6, which improves intra-class similarity between the image signatures.

Various models of BoW have been evaluated in several publications [Yang et al., 2007, Chatfield et al., 2011, Coates and Ng, 2011, Tosic and Frossard, 2011, Boureau et al., 2010a,b]. A recent review of coding schemes [Chatfield et al., 2011] includes Hard Assignment, Soft Assignment, Approximate Locality-constrained Linear Coding, Super Vector Coding, and Fisher Vector Encoding, all evaluated in a common testbed. Furthermore, an insight into the role played by pooling during the generation of image signatures has been studied in [Boureau et al., 2010a,b]. These pooling strategies demonstrated promising improvements in visual categorisation. A detailed comparison of various coding and pooling methods is presented by us in chapter 6, including some improvements in this area. However, none of these evaluations manage to bridge the gap between the classification performance of both groups of BoW introduced above.

To date, the pooling step employed by the first group aggregates only occurrences of visual words in the mid-level features (First-order Occurrence Pooling). In this chapter, we propose to aggregate co-occurrences of visual words in mid-level features to address the second hallmark identified above. Our method is somewhat inspired by Vector of Locally Aggregated Tensors [Negrel et al., 2012] in terms of how we build co-occurrence matrices. However, we distinguish the coding and pooling steps in the proposed model to incorporate arbitrary coders and pooling operators. For the coding step, we employ Sparse Coding (SC), Approximate Locality-constrained Linear Coding (LLC), and Approximate Locality-constrained Soft Assignment (LcSA). This also differs from a recently proposed Second-order Pooling applied in the problem of semantic segmentation [Carreira et al., 2012]: i) we perform pooling on the mid-level features to preserve the data manifold learned during the coding step whilst the latter method acts on the raw descriptors, ii) we provide a derivation of Second- and Higher-order Occurrence Pooling based on linearisation of so-called Minor Polynomial Kernel, iii) a generalised pooling operator is used. Another take on building richer statistics from the mid-level features are $2D$ histogram representations [Yu and Zhang, 2011]. Their work uses various coders and proposes a number of arbitrary statistics for

each of them to retain more information about the coded descriptors. Our approach, however, focuses on capturing co-occurrences as dictated by the analytical solution to a well-defined problem.

To address the third hallmark, we use a generalised pooling operator called @n that was found as a robust performer in chapter 6. The @n can be seen as a trade-off between Max-pooling used by [Yang et al., 2009] and a chosen Analytical pooling, *e.g.* Power Normalisation used in [Perronnin et al., 2010], *theoretical expectation of Max-pooling* proposed in [Boureau et al., 2010b], its approximation AxMin from chapter 6, or the probability of *at least one particular visual word being present in an image* proposed in [Lingqiao et al., 2011]. We opt for @n combined with *at least one particular visual word being present in an image* and simply refer to it as the @n operator in this chapter. Where stated, we use Power Normalisation (Gamma) and *theoretical expectation of Max-pooling* (MaxExp) without the @n scheme, both in their generalised forms to account for the descriptor interdependence, also introduced in chapter 6.

The analysis of First-, Second-, and Third-order Occurrence Pooling in the BoW model constitutes the main contribution of this work. In more detail:

1. We propose to aggregate co-occurrences rather than occurrences of visual words in mid-level features (Second-order Occurrence Pooling).

2. A derivation of Second- and Higher-order Occurrence Pooling based on linearisation of so-called Minor Polynomial Kernel is provided. A generalisation to Average, Max-pooling, and the @n pooling operators is proposed.

3. Simulations show that Second-order Occurrence Pooling is a simple strategy increasing numbers of visual vocabulary elements, thus improving the expressiveness of a given dictionary.

4. Evaluation of First-, Second-, and Third-order Occurrence Pooling is provided for SC, LLC, and LcSA coders. Furthermore, we resign from Spatial Pyramid Matching [Lazebnik et al., 2006, Yang et al., 2009] in favour of Spatial Coordinate Coding proposed in chapter 5, further evaluated in chapter 5, and employed recently by Fisher Vector Encoding in [Sánchez et al., 2012].

5. Moreover, Max-pooling, MaxExp, and the @n pooling operators are compared given SC coder.

6. Second- and Higher-order Occurrence Pooling on bi- and multi-modal codes is proposed based on linearisation of Minor Polynomial Kernel.

7. Evaluation on the grey scale and colour mid-level features is performed for this linearisation and compared to the naive fusing schemes.

8. For further evaluation of the proposed fusion, a residual descriptor is developed to take advantage of the quantisation error yielded by the SC, LLC, and LcSA coders. It is used as a second coder which is complementary to a chosen parent mid-level coder.

9. Spatial Pyramid Matching [Lazebnik et al., 2006] and Dominant Angle Pyramid Matching from chapter 5 are demonstrated as special cases of this fusion.

10. Given various signature sizes, our results are compared in the common testbed to Fisher Vector Encoding (FK) from [Perronnin et al., 2010, Sánchez et al., 2012], Vector of Locally Aggregated Tensors (VLAT) from [Negrel et al., 2012], First-order Occurrence based Spatial Coordinate Coding (SCC), Spatial Pyramid Matching (SPM) [Lazebnik et al., 2006, Yang et al., 2009], and Dominant Angle Pyramid Matching (DoPM). State-of-the-art results are demonstrated on Pascal VOC07, Caltech101, Flower102, and ImageCLEF11 datasets.

Section 7.1.1 introduces the standard model of Bag-of-Words. The coders and pooling operators used in this study are presented in sections 7.1.2 and 7.1.3. Uni-modal BoW with Higher-order Occurrence Pooling is introduced in section 7.2 followed by its derivation sections 7.2.1 and 7.2.2. The benefits of Second-order Occurrence Pooling are illustrated in section 7.2.3. Next, Bi- and Multi-modal BoW with Second- and Higher-order Occurrence Pooling are proposed in section 7.3 followed by their derivations in section 7.3.3. Sections 7.3.1 and 7.3.2 outline the early and late fusion of cues for BoW (used for comparisons on the grey and colour features). Section 7.3.4 presents SPM and DoPM as special cases of our bi-modal fusion. A Residual Descriptor is proposed in

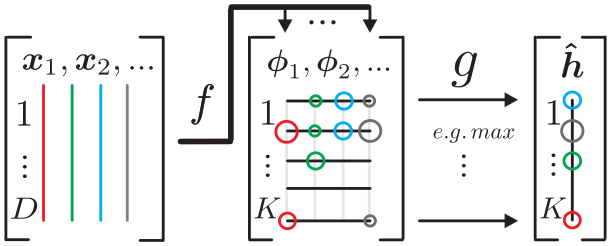


Figure 7.1: Overview of Bag-of-Words. The local descriptors $\boldsymbol{x}$ are extracted from an image and coded by f that operates on columns. Circles of various sizes illustrate values of mid-level coefficients. Pooling g aggregates visual words from the mid-level features $\boldsymbol{\phi}$ along rows.

section 7.3.5 to further demonstrate robustness of the bi-modal fusion. Section 7.4 details the experiments. Uni-modal First-, Second-, and Third-order Occurrence Pooling are compared to FK and VLAT in section 7.4.2. The coding and pooling are evaluated in sections 7.4.3 and 7.4.5. Experiments on Bi-modal Second-order Occurrence Pooling are in section 7.4.4. The final conclusions are drawn in section 7.5.

7.1.1 Bag-of-Words Model

Let us denote the descriptor vectors as $\boldsymbol{x}_n \in \mathbb{R}^D$ such that $n = 1, ..., N$, where N is the total descriptor cardinality for the entire image set $\mathcal{I}$, and D is the descriptor dimensionality. Given any image $i \in \mathcal{I}$, $\mathcal{N}^i$ denotes a set of its descriptor indices. We drop the superscript for simplicity and use $\mathcal{N}$. Therefore, $\{\boldsymbol{x}_n\}_{n \in \mathcal{N}}$ denotes a set of descriptors for an image $i \in \mathcal{I}$. Next, we assume $k = 1, ..., K$ visual appearance prototypes $\boldsymbol{m}_k \in \mathbb{R}^D$ a.k.a. visual vocabulary, words, centres, atoms, and anchors. We form a dictionary $\mathcal{M} = \{\boldsymbol{m}_k\}_{k=1}^K$, where $\mathcal{M} \in \mathbb{R}^{D \times K}$ can also be seen as a matrix formed by visual words as its columns. Figure 7.1 illustrates BoW. Following the notation of chapter 6, the first group of BoW (indicated in the introduction) is a combination of the mid-level coding and pooling steps, followed by the ℓ_2 norm normalisation:

$$\boldsymbol{\phi}_n = f(\boldsymbol{x}_n, \mathcal{M}), \quad \forall n \in \mathcal{N} \tag{7.1}$$

$$\hat{h}_k = g\left(\{\phi_{kn}\}_{n \in \mathcal{N}}\right) \tag{7.2}$$

$$\boldsymbol{h} = \hat{\boldsymbol{h}} / \|\hat{\boldsymbol{h}}\|_2 \tag{7.3}$$

Equation (7.1) represents a chosen mid-level feature mapping $f : \mathbb{R}^D \rightarrow \mathbb{R}^K$, *e.g.* Sparse Coding. It quantifies the image content in terms of the visual vocabulary forming dictionary $\mathcal{M}$. Each descriptor $\boldsymbol{x}_n$ is embedded into the visual vocabulary space resulting in mid-level features $\boldsymbol{\phi}_n \in \mathbb{R}^K$.

Equation (7.2) represents the pooling operation, *e.g.* Average or Max-pooling. The role of g is to aggregate occurrences of visual words in mid-level features, and therefore in an image. Formally, function $g : \mathbb{R}^{|\mathcal{N}|} \rightarrow \mathbb{R}$ takes all mid-level feature coefficients ϕ_{kn} for visual word $\boldsymbol{m}_k$ given image i to produce a k^{th} coefficient in vector $\hat{\boldsymbol{h}} \in \mathbb{R}^K$. Note that $\boldsymbol{\phi}_n$ denotes an n^{th} mid-level feature vector while ϕ_{kn} denotes its k^{th} coefficient. Moreover, we do not assume pooling over cells of Spatial Pyramid Matching to maintain simplicity. SPM compatible formulation can be found in chapter 6.

Equation (7.3) normalises signature $\hat{\boldsymbol{h}}$ to preserve only relative statistics of visual word occurrences in an image, irrespective of the number of descriptors contained within it. The resulting signatures $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^K$ for $i, j \in \mathcal{I}$ form a linear kernel $Ker_{ij} = (\boldsymbol{h}_i)^T \cdot \boldsymbol{h}_j$ and a dual-form KDA classifier [Tahir et al., 2009] is employed.

This model of BoW assumes First-order Occurrence Pooling and often employs SC, LLC, and LcSA coders. However, the same model can accommodate FK and VLAT.

7.1.2 Mid-level coders

Below is the introduction of the mid-level coders f used in this work. For clarity, we abbreviate $\boldsymbol{x}_n$ to $\boldsymbol{x}$ and $\boldsymbol{\phi}_n$ to $\boldsymbol{\phi}$ where possible.

Sparse Coding from [Lee et al., 2007, Yang et al., 2009] expresses each descriptor $\boldsymbol{x}$ as a sparse linear combination of the visual words contained in $\mathcal{M}$. This is achieved by optimising the cost function indicated in equation (6.6), section 6.2.3, chapter 6.

Approximate Locality-constrained Linear Coding addresses the non-locality phenomenon explained in [Wang et al., 2010] that can occur in SC. It optimises the cost function from (6.8), section 6.2.4, chapter 6.

Approximate Locality-constrained Soft Assignment is derived from Mixture of Gaussians given in equation (4.3) from section 4.2 of chapter 4. The membership

probability of component k being selected given descriptor $\boldsymbol{x}$ is further used as a coder $\phi_k = p(k|\boldsymbol{x})$. The coder itself is given in equation (6.9), section 6.2.5, chapter 6.

Fisher Vector Encoding is used in this chapter for comparison purposes. Nonetheless, the coding step can be isolated from its common formulation given in [Perronnin and Dance, 2007, Perronnin et al., 2010]. FK assumes a dictionary based on Gaussian Mixture Model with parameters $\theta = (\theta_1, ..., \theta_K) = ((w_1, \boldsymbol{m}_1, \boldsymbol{\sigma}_1), ..., (w_K, \boldsymbol{m}_K, \boldsymbol{\sigma}_K))$. K denotes the number of Gaussian components, w_k are the component mixing probabilities, $\boldsymbol{m}_k$ are the means, $\boldsymbol{\sigma}_k$ matrices contain on-diagonal standard deviations. The first and second order statistics $\boldsymbol{\psi}_k^{(1)}, \boldsymbol{\psi}_k^{(2)} \in \mathbb{R}^D$ are isolated:

$$\boldsymbol{\psi}_k^{(1)} = \frac{\boldsymbol{x} - \boldsymbol{m}_k}{\boldsymbol{\sigma}_k}, \quad \boldsymbol{\psi}_k^{(2)} = \frac{(\boldsymbol{x} - \boldsymbol{m}_k)^2}{\boldsymbol{\sigma}_k^2} - 1 \tag{7.4}$$

Concatenation of per-cluster statistics $\boldsymbol{\psi}_k \in \mathbb{R}^{2D}$ forms the mid-level feature $\boldsymbol{\phi} \in \mathbb{R}^{2KD}$:

$$\boldsymbol{\phi} = \left[\boldsymbol{\psi}_1^T, ..., \boldsymbol{\psi}_K^T\right]^T, \quad \boldsymbol{\psi}_k = \frac{p\left(k|\boldsymbol{x}, \theta\right)}{\sqrt{w_k}} \begin{bmatrix} \boldsymbol{\psi}_k^{(1)} \\ \frac{1}{\sqrt{2}} \boldsymbol{\psi}_k^{(2)} \end{bmatrix} \tag{7.5}$$

The expression $p\left(k|\boldsymbol{x}, \theta\right)$ is the membership probability of component k being selected given descriptor $\boldsymbol{x}$ and parameters θ. Note that the above formulation is compatible with equation 7.1 except for $\boldsymbol{\phi}$ to be $2KD$ rather than K long. Moreover, the resulting $\boldsymbol{\phi}$ for FK contains second-order statistics unlike $\boldsymbol{\phi}$ that is generated by the SC, LLC, and LcSA coders.

Vector of Locally Aggregated Tensors from [Negrel et al., 2012] also has a distinct coding step yielding the first and second order statistics $\boldsymbol{\psi}_k^{(1)} \in \mathbb{R}^D$ and $\boldsymbol{\Psi}_k^{(2)} \in \mathbb{R}^{D \times D}$ per cluster:

$$\boldsymbol{\Psi}_k^{(2)} = \boldsymbol{\psi}_k^{(1)} \boldsymbol{\psi}_k^{(1)^T} - \boldsymbol{C}_k, \quad \boldsymbol{\psi}_k^{(1)} = \boldsymbol{x} - \boldsymbol{m}_k \tag{7.6}$$

However, only the second order matrices $\boldsymbol{\Psi}_k^{(2)}$ are deployed to form the mid-level features after normalisation with per-cluster covariance matrices $\boldsymbol{C}_k$. As $\boldsymbol{\Psi}_k^{(2)}$ are symmetric, the upper triangles and diagonals are extracted and flattened into vectors with operator $u_:$, and concatenated for all k-means clusters $k = 1, ..., K$:

$$\boldsymbol{\phi} = \left[u_:(\boldsymbol{\Psi}_1^{(2)})^T, ..., u_:(\boldsymbol{\Psi}_K^{(2)})^T\right]^T, \tag{7.7}$$

Note that VLAT is also compatible with equation 7.1 except for $\boldsymbol{\phi}$ to be $KD(D+1)/2$ rather than K long.

7.1.3 Pooling Operators

BoW introduced in section 7.1.1 assumes aggregation of occurrences of visual words represented by the coefficients of mid-level feature vectors with a pooling operator g given by equation (7.2). It was demonstrated in chapter 6 that the choice of pooling influences the classification performance of various coders. The operators used in this work are briefly described below.

Average pooling counts the number of descriptor assignments per cluster k and normalises such counts by the number of descriptors in the image [Csurka et al., 2004, van Gemert et al., 2008, 2010]. However, it can also work with various coders, *e.g.* SC, LLC, LcSA, FK, VLAT. It is expressed as:

$$\hat{h}_k = \text{avg}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) = \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}}\phi_{kn} \tag{7.8}$$

Max-pooling forms image signatures from the largest coefficients per visual word [Yang et al., 2009, Boureau et al., 2010a,b, Lingqiao et al., 2011], thus the largest value between $|\mathcal{N}|$ mid-level features responding to visual word $\boldsymbol{m}_k$ is selected:

$$\hat{h}_k = \max\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) = \max\left(\phi_{kn^{(1)}}, \phi_{kn^{(2)}}, ...\right) \tag{7.9}$$

Max-pooling is often combined with SC, LLC, or LcSA rather than FK or VLAT.

MaxExp operator is a likelihood inspired *theoretical expectation of Max-pooling* proposed in [Boureau et al., 2010b], described in section 6.3.2, and expressed as:

$$\hat{h}_k = 1 - (1 - h_k^*)^{\bar{N}}, \quad h_k^* = \text{avg}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \tag{7.10}$$

Moreover, we generalised this operator to account for the feature interdependence in section 6.3.4. As the degree of statistical dependence between features is unknown, parameter $\bar{N} \leq |\mathcal{N}|$ has to be found by cross-validation. MaxExp is typically used with SC, LLC, and LcSA because inequality $0 \leq h_k^* \leq 1$ has to hold. FK and VLAT may violate this constraint.

Power Normalisation (Gamma) used in [Boughorbel et al., 2005, Perronnin et al., 2010, Jégou et al., 2009] was shown to be closely related to MaxExp in section 6.17. It

also follows a close probabilistic interpretation to MaxExp. A generalised form for FK and VLAT that aggregates over positive-negative ϕ_{kn} is given as:

$$\hat{h}_k = \operatorname{sgn}(h_k^*)\,|h_k^*|^\gamma\,, \quad h_k^* = \operatorname{avg}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \tag{7.11}$$

The correction factor $0<\gamma\leq 1$ accounts for the degree of statistical dependence between features and is found by cross-validation. Gamma also works with SC, LLC, and LcSA.

Improved pooling (@n) was proposed in section 6.3.5. It is designed to suppress the low values of mid-level feature coefficients that were recognised as a noise and called *leakage*. Given SC, LLC, and LcSA coders, *leakage* was shown to misrepresent chosen visual prototypes. Moreover, the @n was shown to exploit the descriptor interdependence and led to consistent classification improvements. Such an operator is a trade-off between Max-pooling and a chosen Analytical pooling, *e.g.* MaxExp:

$$\begin{aligned} h_k^* &= \operatorname{avg}\operatorname{srt}\left(\{\phi_{kn}\}_{n\in\mathcal{N}},@n\right)=\operatorname{avg}\left[\operatorname{srt}\left(\{\phi_{kn}\}_{n\in\mathcal{N}},@n\right)\right] \\ \hat{h}_k &= 1-(1-h_k^*)^{\bar{N}} \end{aligned} \tag{7.12}$$

The @n largest mid-level features are selected by partial sort algorithm srt and averaged by *avg*. Parameter $1\leq @n\leq|\mathcal{N}|$ adjusts the trade-off, while meaning of $\bar{N}$ remains the same as for MaxExp. The mid-level feature coefficients for any given $\boldsymbol{m}_k$ are presumed to be drawn at random from a Bernoulli distribution under the i.i.d. assumption. However, this is only approximately true as ϕ_{kn} are typically non-negative real numbers such that $0\leq\phi_{kn}\leq 1$. Note that the pool of the largest @n coefficients only is available. Binomial distribution dictates that, given exactly $\bar{N}=@n$ trials, equation (7.12) yields the probability of *at least one visual word* $\boldsymbol{m}_k$ *present in the* @n *largest mid-level feature coefficients*. Given that the largest coefficients represent visual word $\boldsymbol{m}_k$ and the smaller ones the noise, this formulation yields improved estimates. However, this assumption does not directly apply to FK or VLAT.

An analytical trade-off between Average and Max-pooling similar to partially sorting and averaging the mid-level features can be also obtained with the ℓ_p norm. This combined with MaxExp results in an alternative operator (MaxExp+ℓ_p):

$$\hat{h}_k = 1-(1-h_k^*)^{\bar{N}}\,, \quad h_k^* = \left(\frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}}|\phi_{kn}|^p\right)^{1/p} \tag{7.13}$$

The solution between Average and Max-pooling is varied by $1\leq p\leq\infty$ instead of @n.

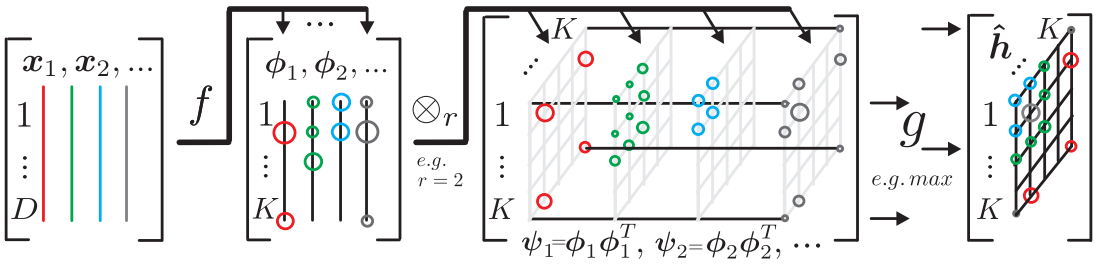


Figure 7.2: Uni-modal Bag-of-Words with Second-order Occurrence Pooling (order $r = 2$). The local descriptors $\boldsymbol{x}$ are extracted from an image and coded by f that operates on columns. Circles of various sizes illustrate values of mid-level coefficients. Self-tensor product $\otimes_r$ computes co-occurrences of visual words for every mid-level feature $\boldsymbol{\phi}$. Pooling g aggregates visual words from the co-occurrence matrices $\boldsymbol{\psi}$ along the direction of stacking. For the purpose of illustration, the flattening operator $u_:$ from equation (7.15) is not used.

7.2 Uni-modal Bag-of-Words with Higher-Order Occurrence Pooling

Section 7.1.1 gave an overview of Bag-of-Words with First-order Occurrence Pooling, typically using the reviewed coding and pooling operators. However, FK and VLAT were demonstrated to benefit from the second-order statistics. To equip BoW in the second- or higher-order statistics, we re-formulate it:

$$\boldsymbol{\phi}_n = f(\boldsymbol{x}_n, \mathcal{M}), \quad \forall n \in \mathcal{N} \tag{7.14}$$

$$\boldsymbol{\psi}_n = u_: (\otimes_r \boldsymbol{\phi}_n) \tag{7.15}$$

$$\hat{h}_k = g\left(\{\psi_{kn}\}_{n \in \mathcal{N}}\right) \tag{7.16}$$

$$\boldsymbol{h} = \hat{\boldsymbol{h}} / \|\hat{\boldsymbol{h}}\|_2 \tag{7.17}$$

A relevant derivation will follow in section 7.2.1, but first we explain how the proposed extension differs from BoW given in section 7.1.1. Figure 7.2 illustrates Bag-of-Words with Second-order Occurrence Pooling in contrast to the typical BoW in figure 7.1.

Equation (7.14) represents a chosen mid-level feature mapping $f : \mathbb{R}^D \to \mathbb{R}^K$, *e.g.* SC, LLC, or LcSA.

Equation (7.15) represents tensor self-product $\otimes_r$ performed on every mid-level feature vector $\boldsymbol{\phi}_n$ resulting from f, where $r \geq 1$ is a chosen rank (or order). This is done in order to compute co-occurrences (or higher-order occurrences) of visual words in every mid-level feature. Given $r = 1$, the above formulation becomes reduced to the standard BoW as $\boldsymbol{\psi}_n = \boldsymbol{\phi}_n = \otimes_1 (\boldsymbol{\phi}_n)$. Moreover, as the resulting $\otimes_{r>1}$ are symmetric, only non-redundant coefficients are retained and flattened into vectors with operator $u_:$. Specifically, one can extract: i) the upper triangle and diagonal for $\otimes_2$, ii) the upper pyramid and diagonal plane for $\otimes_3$, iii) the upper simplex and diagonal hyperplane for $\otimes_{r\geq 3}$. The dimensionality of self-tensor product after rejecting repeated coefficients and flattening is $K^{(r)} = \binom{K+r-1}{r}$.

Equation (7.16) represents the pooling operation, as in sections 7.1.1 and 7.1.3. However, this time g aggregates co-occurrences or higher-order occurrences of visual words in mid-level features for $r = 2$ or $r > 2$, respectively. Formally, function $g : \mathbb{R}^{|\mathcal{N}|} \rightarrow \mathbb{R}$ takes k^{th} co-occurrence (or higher-order occurrence) coefficients ψ_{kn} for all $n \in \mathcal{N}$ given image i to produce a k^{th} coefficient in vector $\hat{\boldsymbol{h}} \in \mathbb{R}^{K^{(r)}}$, where $k = 1, ..., K^{(r)}$.

Equation (7.17) is the normalisation step performed on $\hat{\boldsymbol{h}}$ to preserve only the relative statistics of visual word co-occurrences (or higher-order occurrences). Note that the resulting signatures $\boldsymbol{h}$ are of dimensionality $K^{(r)}$ which depends on the dictionary size K and rank r, and remains independent of the descriptor dimensionality D. This is in contrast to sizes of FK and VLAT signatures depending on both K and D.

7.2.1 Linearisation of Minor Polynomial Kernel

The proposed BoW with Higher-order Occurrence Pooling can be derived analytically by performing the following steps: i) defining a kernel function on a pair of mid-level features and referred to as Minor Kernel, ii) summing over all pairs of mid-level features formed from two images, iii) normalising with respect to the feature count and, iv) normalising the resulting kernel. First, we define Minor Polynomial Kernel:

$$ker\left(\boldsymbol{\phi}, \bar{\boldsymbol{\phi}}\right) = \left(\beta \boldsymbol{\phi}^T \bar{\boldsymbol{\phi}} + \lambda\right)^r \tag{7.18}$$

We chose $\beta = 1$ and $\lambda = 0$, while $r \geq 1$ denotes the polynomial degree (it is also the order of occurrence pooling). Equation (7.18) can be rewritten by using the dot product

$\langle \boldsymbol{\phi}, \bar{\boldsymbol{\phi}} \rangle$ of a pair of mid-level features:

$$ker\left(\boldsymbol{\phi}, \bar{\boldsymbol{\phi}}\right) = \langle \boldsymbol{\phi}, \bar{\boldsymbol{\phi}} \rangle^r \tag{7.19}$$

We assume $\boldsymbol{\phi}$ and $\bar{\boldsymbol{\phi}}$ are the ℓ_2 norm normalised. We define a kernel function between two sets of mid-level features $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_n\}_{n\in\mathcal{N}}$ and $\bar{\boldsymbol{\Phi}} = \{\bar{\boldsymbol{\phi}}_{\bar{n}}\}_{\bar{n}\in\bar{\mathcal{N}}}$ given two sets of descriptor indexes $\mathcal{N}$ and $\bar{\mathcal{N}}$ from two images:

$$\begin{aligned} Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) &= \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}}\frac{1}{|\bar{\mathcal{N}}|}\sum_{\bar{n}\in\bar{\mathcal{N}}} ker\left(\boldsymbol{\phi}_n, \bar{\boldsymbol{\phi}}_{\bar{n}}\right) \\ &= \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}}\frac{1}{|\bar{\mathcal{N}}|}\sum_{\bar{n}\in\bar{\mathcal{N}}} \langle \boldsymbol{\phi}_n, \bar{\boldsymbol{\phi}}_{\bar{n}} \rangle^r \\ &= \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}}\frac{1}{|\bar{\mathcal{N}}|}\sum_{\bar{n}\in\bar{\mathcal{N}}} \left(\sum_{k=1}^{K} \phi_{kn}\bar{\phi}_{k\bar{n}}\right)^r \end{aligned} \tag{7.20}$$

Moreover, the rightmost summation in equation (7.20) can be re-expressed as a dot product of two self-tensor products of order r. Similar considerations were previously shown in [Picard and Gosselin, 2011]. Thus, this leads to:

$$\begin{aligned} \left(\sum_{k=1}^{K} \phi_{kn}\bar{\phi}_{k\bar{n}}\right)^r &= \sum_{k^{(1)}=1}^{K} \dots \sum_{k^{(r)}=1}^{K} \phi_{k^{(1)}}\bar{\phi}_{k^{(1)}} \cdot \dots \cdot \phi_{k^{(r)}}\bar{\phi}_{k^{(r)}} \\ &= \left\langle u_{:}^{*}(\otimes_r \boldsymbol{\phi}_n), u_{:}^{*}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right)\right\rangle \end{aligned} \tag{7.21}$$

Operator $u_{:}^{*}$ is used to flatten an r dimensional tensor into a vector. Now, equation (7.20) is further simplified:

$$\begin{aligned} Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) &= \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}}\frac{1}{|\bar{\mathcal{N}}|}\sum_{\bar{n}\in\bar{\mathcal{N}}} \left\langle u_{:}^{*}(\otimes_r \boldsymbol{\phi}_n), u_{:}^{*}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right)\right\rangle \\ &= \left\langle \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}} u_{:}^{*}(\otimes_r \boldsymbol{\phi}_n), \frac{1}{|\bar{\mathcal{N}}|}\sum_{\bar{n}\in\bar{\mathcal{N}}} u_{:}^{*}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right)\right\rangle \\ &= \left\langle \underset{n\in\mathcal{N}}{\text{avg}}\left[u_{:}^{*}(\otimes_r \boldsymbol{\phi}_n)\right], \underset{\bar{n}\in\bar{\mathcal{N}}}{\text{avg}}\left[u_{:}^{*}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right)\right]\right\rangle \end{aligned} \tag{7.22}$$

We denote $\text{avg}_{n\in\mathcal{N}}\, \boldsymbol{v}_n$ as a mean vector over $\{\boldsymbol{v}_n\}_{n\in\mathcal{N}}$. Moreover, kernel $Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right)$ is normalised to ensure that self-similarity $Ker\left(\boldsymbol{\Phi}, \boldsymbol{\Phi}\right) = Ker\left(\bar{\boldsymbol{\Phi}}, \bar{\boldsymbol{\Phi}}\right) = 1$. This is achieved by applying a well-known formula:

$$Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) := \frac{Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right)}{\sqrt{Ker\left(\boldsymbol{\Phi}, \boldsymbol{\Phi}\right)}\sqrt{Ker\left(\bar{\boldsymbol{\Phi}}, \bar{\boldsymbol{\Phi}}\right)}} \tag{7.23}$$

Therefore, the model in equations (7.20) and (7.22) can be readily re-expressed in terms of generalised equations (7.14), (7.15) and (7.16) from section 7.2 if g in equation (7.16) is defined as Average pooling from equation (7.8). We also replace the flattening operator $u^*_:$ with previously defined $u_:$ to reject the redundant coefficients from the symmetric self-tensor products.

7.2.2 Beyond Average Pooling for Higher-order Occurrence Statistics

It has been demonstrated in several evaluations on the visual categorisation tasks that Average pooling performs worse than Max-pooling [Yang et al., 2009, Boureau et al., 2010b]. This can be explained by stressing that Average pooling counts occurrences of any given visual prototype in an image. Therefore, it quantifies areas spanned by repeatable patterns that are unlikely to appear in the same quantities in a collection of images. However, Max-pooling was shown to be a lower bound of the likelihood of *at least one visual word* $\boldsymbol{m}_k$ *being present in image* i [Lingqiao et al., 2011]. Thus, Max-pooling acts largely as a detector of visual prototypes and delivers better classification results. Below we provide a generalisation of Higher-order Occurrence Pooling to work with Max-pooling and the @n operator to benefit the classification process.

First, we assume two sets of mid-level features $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_n\}_{n\in\mathcal{N}}$ and $\bar{\boldsymbol{\Phi}} = \{\bar{\boldsymbol{\phi}}_{\bar{n}}\}_{\bar{n}\in\bar{\mathcal{N}}}$ and their descriptor indexes $\mathcal{N}$ and $\bar{\mathcal{N}}$ from two images. We also define $\max_{n\in\mathcal{N}} v_n = \max(\{v_n\}_{n\in\mathcal{N}})$ and $\max_{n\in\mathcal{N}} \boldsymbol{v}_n$ as a vector formed from element-wise $\max(\{v_{1n}\}_{n\in\mathcal{N}})$, $\max(\{v_{2n}\}_{n\in\mathcal{N}})$, ..., applied over all $\boldsymbol{v}_n$.

The standard BoW with Max-pooling and Polynomial Kernel of degree r is given in equation (7.24) which is then expanded in equation (7.25) and simplified to a dot product between two vectors in equation (7.26). Such an expression forms a linear kernel. A simple lower bound of equation (7.25) is proposed in equation (7.27). Note that it represents Higher-order Occurrence Pooling with Max-pooling operator further

linearised to a dot product between two vectors in equation (7.28).

$$Ker\left(\boldsymbol{\Phi},\bar{\boldsymbol{\Phi}}\right)=\left\langle \hat{\boldsymbol{h}},\bar{\hat{\boldsymbol{h}}}\right\rangle^{r},\ \text{and}\ \begin{cases}\hat{h}_k=\max\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right)\\ \bar{\hat{h}}_k=\max\left(\{\bar{\phi}_{kn}\}_{\bar{n}\in\bar{\mathcal{N}}}\right)\end{cases}$$

$$=\left(\sum_{k=1}^{K}\max_{n\in\mathcal{N}}(\phi_{kn})\cdot\max_{\bar{n}\in\bar{\mathcal{N}}}(\bar{\phi}_{k\bar{n}})\right)^{r} \tag{7.24}$$

$$=\sum_{k^{(1)}=1}^{K}\ldots\sum_{k^{(r)}=1}^{K}\left(\begin{array}{r}\max_{n\in\mathcal{N}}(\phi_{k^{(1)}n})\cdot\ldots\cdot\max_{n\in\mathcal{N}}(\phi_{k^{(r)}n})\cdot\\ \cdot\max_{\bar{n}\in\bar{\mathcal{N}}}(\bar{\phi}_{k^{(1)}\bar{n}})\cdot\ldots\cdot\max_{\bar{n}\in\bar{\mathcal{N}}}(\bar{\phi}_{k^{(r)}\bar{n}})\end{array}\right) \tag{7.25}$$

$$=\left\langle u_{:}^{*}\left[\,\otimes_r\max_{n\in\mathcal{N}}\left(\boldsymbol{\phi}_n\right)\right],u_{:}^{*}\left[\,\otimes_r\max_{\bar{n}\in\bar{\mathcal{N}}}\left(\bar{\boldsymbol{\phi}}_{\bar{n}}\right)\right]\right\rangle \tag{7.26}$$

$$\geq\sum_{k^{(1)}=1}^{K}\ldots\sum_{k^{(r)}=1}^{K}\left(\begin{array}{r}\max_{n\in\mathcal{N}}(\phi_{k^{(1)}n}\cdot\ldots\cdot\phi_{k^{(r)}n})\cdot\\ \cdot\max_{\bar{n}\in\bar{\mathcal{N}}}(\phi_{k^{(1)}\bar{n}}\cdot\ldots\cdot\bar{\phi}_{k^{(r)}\bar{n}})\end{array}\right) \tag{7.27}$$

$$=\left\langle\max_{n\in\mathcal{N}}\left[u_{:}^{*}(\otimes_r\boldsymbol{\phi}_n)\right],\max_{\bar{n}\in\bar{\mathcal{N}}}\left[u_{:}^{*}\left(\otimes_r\bar{\boldsymbol{\phi}}_{\bar{n}}\right)\right]\right\rangle \tag{7.28}$$

This lower bound emerges from breaking bi-linearity of Average pooling pooling by applying Max-pooling. The formulation from section 7.2.1 helps predict the structure of the linearised equations, but its performance remains identical to the parent method. Breaking bi-linearity in Negrel et al. [2012] led to large improvements over its bi-linearity preserving equivalent in Picard and Gosselin [2011]. We argue that breaking bi-linearity in the pooling step is essential for improving results of the higher-order methods. We observed that the lower bound formulations result in signatures having lower normalised entropy compared to the parent methods. Thus, they preserve more refined information about each image. We also verified this analytically for $K=2$ and $r=2$.

Next, an interesting probabilistic difference between models in equations (7.26) and (7.28) can be shown. Let us consider Max-pooling in the standard BoW model (without Polynomial Kernel). If mid-level feature coefficients ϕ_{kn} are drawn from a feature distribution under the i.i.d. assumption given a visual word $\boldsymbol{m}_k$, the likelihood of *at least one visual word* $\boldsymbol{m}_k$ *being present in image* i [Lingqiao et al., 2011] can be expressed as follows:

$$1-\prod_{n\in\mathcal{N}}(1-\phi_{kn})\geq\max\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \tag{7.29}$$

Note that such a probability is an upper bound of Max-pooling. Furthermore, one can derive upper bounds of Max-pooling for the problems in equations (7.26) and (7.28). We denote the not-flattened image signature from equation (7.26) as tensor $\boldsymbol{T} = \otimes_r \max_{n \in \mathcal{N}} (\boldsymbol{\phi}_n) \in \mathbb{R}^{K^r}$. Coefficient-wise, this can be expressed as:

$$T_{k^{(1)},\dots,k^{(r)}} = \prod_{s=1}^{r} \max\left(\{\phi_{k^{(s)}n}\}_{n \in \mathcal{N}}\right) \tag{7.30}$$

Note that every coefficient of image signature of Bag-of-Words with Max-pooling and Polynomial Kernel is upper bounded by the probability of *visual words* $\boldsymbol{m}_{k^{(1)}}, \dots, \boldsymbol{m}_{k^{(r)}}$ *jointly occurring at least once in image* i:

$$\prod_{s=1}^{r} \left(1 - \prod_{n \in \mathcal{N}} (1 - \phi_{k^{(s)}n})\right) \geq T_{k^{(1)},\dots,k^{(r)}} \tag{7.31}$$

Moreover, we denote the not-flattened image signature from equation (7.28) as tensor $\boldsymbol{T}' = \max_{n \in \mathcal{N}} \left(\otimes_r \boldsymbol{\phi}_n\right) \in \mathbb{R}^{K^r}$. Coefficient-wise, this can be expressed as:

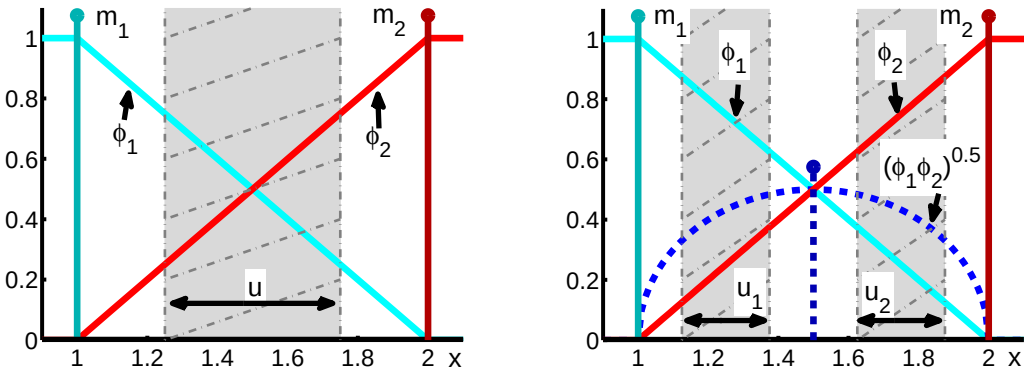
$$T'_{k^{(1)},\dots,k^{(r)}} = \max\left(\left\{\prod_{s=1}^{r} \phi_{k^{(s)}n}\right\}_{n \in \mathcal{N}}\right) \tag{7.32}$$

Again, we note that every coefficient of image signature of Higher-order Occurrence Pooling with Max-pooling operator is upper bounded by the probability of *visual words* $\boldsymbol{m}_{k^{(1)}}, \dots, \boldsymbol{m}_{k^{(r)}}$ *jointly occurring in at least one mid-level feature* $\boldsymbol{\phi}_n$:

$$1 - \prod_{n \in \mathcal{N}} \left(1 - \prod_{s=1}^{r} \phi_{k^{(s)}n}\right) \geq T'_{k^{(1)},\dots,k^{(r)}} \tag{7.33}$$

The joint occurrence of visual words on the mid-level feature level expressed in equation (7.33) is more informative compared to the joint occurrence on the image level in equation (7.31) as, it can be thought of as adding new elements to the visual dictionary. This will be demonstrated in the next section.

In practice, we use Second- and Higher-order Occurrence Pooling with the @n operator. Under minor changes, the standard BoW model with the @n operator and Polynomial Kernel is shown to be an upper bound of such a model. See appendix A.3 for details. Furthermore, applying normalisation from equation (7.23) is equivalent to the ℓ_2 norm normalising the image signatures. Lastly, the operator $u_:$ defined earlier is used in place of $u^*_:$ to reject the redundant coefficients from the symmetric self-tensor products.



(*a*) first-order (*b*) second-order

Figure 7.3: Uncertainty in Max-pooling. Mid-level feature coefficients ϕ_1 and ϕ_2 are produced by LLC ($l=2$) for descriptors $x \in \langle 1;2 \rangle$ given visual words $m_1=1$ and $m_2=2$. (*a*) First-order Occurrence Pooling results in the pooling uncertainty u (the grey area). See text for explanations. (*b*) Second-order statistics produce co-occurrence component $(\phi_1\phi_2)^{0.5}$ that has a maximum for x indicated by the dashed stem. This component limits the pooling uncertainty. The square root is applied to preserve the linear slopes, *e.g.* $(\phi_1\phi_1)^{0.5}=\phi_1$.

7.2.3 Interpretation of the Joint Occurrence of Visual Words on the Mid-level Feature Level

This section provides intuitive considerations on Second-order Occurrence Pooling. We argue that the joint occurrence of visual words on the mid-level feature level benefits Max-pooling (and related operators) by limiting its pooling uncertainty as detailed below.

Figure 7.3 illustrates the mid-level coefficients produced with LLC ($l=2$) for descriptors $x \in \langle 1;2 \rangle$. Two one dimensional visual words are used.

Figure 7.3 (a) shows two linear slopes comprised of coding values ϕ_1 and ϕ_2 for any $1 \leq x \leq 2$. Imagine that we draw randomly a number of descriptors from this interval, obtain ϕ_1 and ϕ_2 from the plot, and apply Max-pooling. Note that the role of pooling is to aggregate the mid-level features into an image signature and preserve information about the descriptors. If we were to draw several times $x_n = 1.5$, we would obtain $\phi_{1n} = \phi_{2n} = 0.5$ for all n. Applying Max-pooling would result in $max(\{\phi_{1n}\}_{n \in \mathcal{N}}) =$

$max(\{\phi_{2n}\}_{n\in\mathcal{N}}) = 0.5$. From this information, one can infer that the only descriptors that could produce such signature are $x_n = 0.5$. Therefore, if $max(\{\phi_{1n}\}_{n\in\mathcal{N}}) \to 0.5$ and $max(\{\phi_{2n}\}_{n\in\mathcal{N}}) \to 0.5$, uncertainty in position of descriptors x_n results in $u \to 0$. However, it takes only two descriptors $x_1 = 1$ and $x_2 = 2$ to mask presence of other descriptors x such that $1 < x < 2$. In this case, Max-pooling results in $max(\{\phi_{1n}\}_{n\in\mathcal{N}}) = max(\{\phi_{2n}\}_{n\in\mathcal{N}}) = 1$. One can infer that $x_1 = 1$ and $x_2 = 2$ were present amongst the descriptors. However, other descriptors $1 < x < 2$ could have been also present, *e.g.* $x_3 = 1.25$, $x_4 = 1.5$, and $x_5 = 1.75$. However, there is nothing in the produced signature indicating this. Thus, as $max(\{\phi_{1n}\}_{n\in\mathcal{N}}) \to 1$ and $max(\{\phi_{2n}\}_{n\in\mathcal{N}}) \to 1$, uncertainty in position of descriptors x_n results in $u \to 1$. Both these cases seem undesirable, *e.g.* if all $x_n = 1.5$ then there are no other descriptors in the image. If $x_1 = 1$ and $x_2 = 2$ then another descriptors are masked during Max-pooling.

Figure 7.3 (b) extends the above experiment with the second-order statistics. Co-

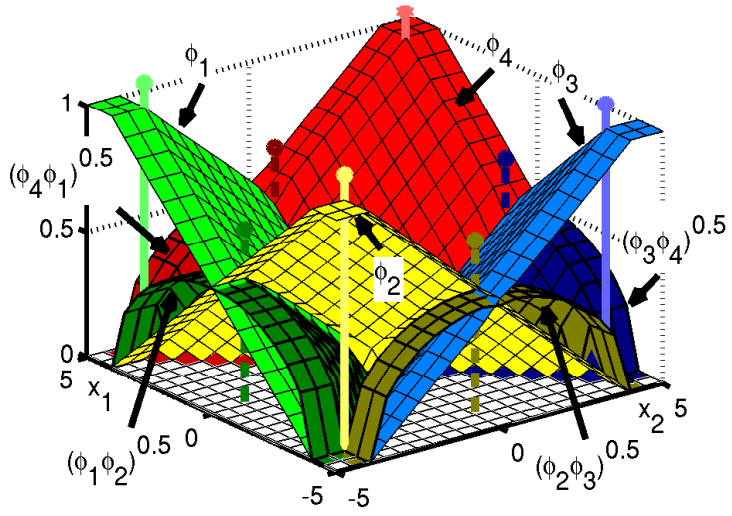
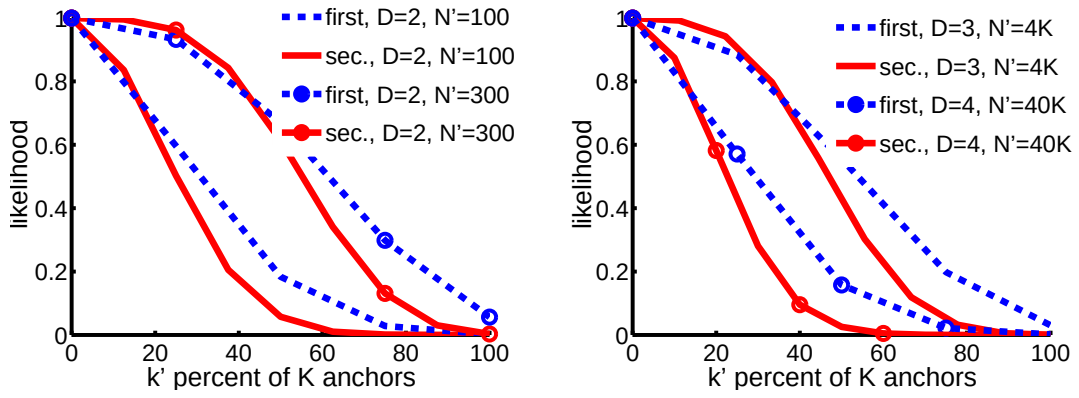


Figure 7.4: Co-occurrence coefficients. Mid-level feature coefficients $\phi_1, ..., \phi_4$ are produced by SC ($\alpha = 1$) for descriptors $\boldsymbol{x} = [x_1, x_2]^T \in \langle -5; 5\rangle^2$ and arbitrarily chosen $k = 1, ..., 4$ visual words $\boldsymbol{m}_k \in \langle -5; 5\rangle^2$ indicated by the solid line stems. The second-order statistics produce co-occurrence components $(\phi_1\phi_2)^{0.5}$, $(\phi_2\phi_3)^{0.5}$, $(\phi_3\phi_4)^{0.5}$, and $(\phi_4\phi_1)^{0.5}$ with maxima for $\boldsymbol{x}$ indicated by the dashed stems. The remaining co-occurrence coefficients are equal 0, *e.g.* $(\phi_1\phi_3)^{0.5} = 0$. This shows that the subspace learned with SC is preserved.

occurrence of ϕ_1 and ϕ_2 results in coefficient $\phi_1\phi_2$. We applied the square root to these statistics to preserve the linear slopes of ϕ_1 and ϕ_2 in the plot, *e.g.* we plotted $(\phi_1\phi_2)^{0.5}$ as a dashed curve instead of $\phi_1\phi_2$. We indicated that the maximum of this function is attained for descriptor $x=1.5$ (the dashed stem). If two descriptors $x_1=1$ and $x_2=2$ are drawn, this time they cannot fully mask other descriptors x such that $1<x<2$. Max-pooling for these descriptors results in $max(\{\phi_{1n}\}_{n\in\mathcal{N}})=max(\{\phi_{2n}\}_{n\in\mathcal{N}})=1$ and $max(\{\phi_{1n}\phi_{2n}\}_{n\in\mathcal{N}})=0$. Note that drawing a third descriptor $x_3=1.5$ would result in $max(\{\phi_{1n}\phi_{2n}\}_{n\in\mathcal{N}})=0.5$ bearing its mark in the image signature. Hence, we consider the second-order statistics to be a simple approach that increases resolution of a visual dictionary. This limits the uncertainty of Max-pooling such that $u_1+u_2\leq u$.

Figure 7.4 illustrates the mid-level coefficients ϕ_1, ϕ_2, ϕ_3, ϕ_4 produced with SC ($\alpha=1$) for $\boldsymbol{x}=[x_1,x_2]^T\in\langle-5;5\rangle^2$, and the corresponding co-occurrence coefficients $(\phi_1\phi_2)^{0.5}$, $(\phi_2\phi_3)^{0.5}$, $(\phi_3\phi_4)^{0.5}$, $(\phi_4\phi_1)^{0.5}$. We applied the square root to these statistics to preserve the linear slopes of ϕ_1, ϕ_2, ϕ_3, and ϕ_4. The maxima of the co-occurrence functions are indicated by the dashed stems. They can be seen as the additional elements of the visual dictionary. Note that $(\phi_1\phi_3)^{0.5}=(\phi_2\phi_4)^{0.5}=0$ for any $\boldsymbol{x}\in\langle-5;5\rangle^2$. This demonstrates



(a) (b)

Figure 7.5: The saturation effect in Max-pooling for the first- and second-order pooling ('first' and 'sec'). Descriptor space $\langle-5;5\rangle^D$ is quantised into 21^D values. We draw from it N' values given the uniform distribution. (a) Likelihood that at least k' percent of $K=4$ anchors will overlap with N' descriptors given $D=2$. (b) Simulation for $D=3$ and $D=4$. Note that the second-order pooling exhibits less saturation in all cases.

that the subspace learned with SC is preserved in the second-order statistics in contrast to $2D$ histogram representations [Yu and Zhang, 2011] that compute sum between all pairs of mid-level feature coefficients. We applied summing and noted it yielded worse results compared to the co-occurrence statistics. We also tried to bypass the coding step as Second-order Pooling in [Carreira et al., 2012]. This also yielded lower results than the proposed co-occurrence statistics on mid-level features. These observations indicate the importance of subspace/manifold learning with sparse coding techniques.

We illustrated earlier that if the descriptors overlap with the anchors from the dictionary, the remaining descriptors are not represented in the final signature. Therefore, we perform an experiment to quantify this behaviour. Figure 7.5 illustrates likelihood that at least k' percent of $K=4$ anchors will overlap with N' descriptors that are drawn at random from descriptor space $\langle -5; 5\rangle^D$ quantised into 21^D values. We consider an anchor to overlap with a descriptor if their both quantised values are the same. Figure 7.5 (a) demonstrates that if $N'=300$ descriptors are drawn given $D=2$, it is 5% likely they will overlap with all 4 anchors. For $N'=100$ descriptors this is unlikely. Furthermore, the second-order statistics contribute additional 4 non-zero coefficients that increase resolution of the visual dictionary (see figure 7.4). Therefore, it is more likely that for the second-order cases, the descriptors will overlap with at least one anchor more likely than for the first-order cases. However, it is less likely that the descriptors will overlap with all anchors for the second-order cases compared to the first-order representations. This demonstrates that the second-order statistics improve capabilities of Max-pooling (and related pooling operators). Lastly, figure 7.5 (b) demonstrates the same behaviour in higher dimensional spaces as, for $D=3$ and $D=4$, there are 5 and 6 non-zero second-order coefficients, respectively.

7.3 Bag-of-Words for Bi- and Multi-modal Codes with Second- and Higher-Order Occurrence Pooling

Grey scale and colour cues are often combined due to their complementary nature that benefits the object category recognition [van de Sande et al., 2008, Perronnin et al., 2010, Nilsback and Zisserman, 2008b,a, Yuan and Yan, 2010, Bosch et al., 2007, Yang

et al., 2012a, Yan et al., 2010] and visual concept detection [Nowak et al., 2011, Huiskes and Lew, 2008, Tahir et al., 2010, Binder et al., 2011, Su and Jurie, 2011]. Some approaches employ so-called early fusion of modalities that occurs on the descriptor level, *e.g.* [van de Sande et al., 2008] and descriptors from chapter 3. Another methods perform coding and pooling steps on various modalities first, followed by so-called late fusion which involves combining multiple kernels [Nilsback and Zisserman, 2008b,a, Yang et al., 2012a, Yan et al., 2010, Koniusz and Mikolajczyk, 2010, Tahir et al., 2010].

The Second- and Higher-order Occurrence Pooling are characterised by their ability to capture the joint occurrence of visual words per mid-level feature as formulated in equation (7.33) of section 7.2.2. This ability extends to bi- and multi-modal scenarios. Each modality represented by mid-level features of some kind results in the joint occurrence statistics. Furthermore, linearisation of Minor Polynomial Kernel predicts existence of a cross-term which can be characterised as the joint occurrence of visual words between various kinds of mid-level features, *e.g.* grey and colour features that correspond to each other spatially-wise.

We first formulate the typical early and late fusion approaches that are used for comparisons in this chapter, followed by derivation of Bi-modal Second- and Higher-order Occurrence Pooling based on linearisation of Minor Polynomial Kernel.

7.3.1 Early Fusion in Bag-of-Words

We showed in chapter 5 that the early fusion of modalities can be thought of as a trade-off between the quantisation losses of linearly coded signals. With the means of Sparse Coding, we showed that such a trade-off can be implemented by concatenating modalities on the descriptor level without explicitly redesigning the coding method. Such a fusion of descriptors with their spatial coordinates is called Spatial Coordinate Coding, which was introduced in section 5.2. It improves the classification performance and limits the size of image signatures due to bypassed Spatial Pyramid Matching [Lazebnik et al., 2006]. A similar fusion on descriptor level was also used in recognition with discriminatively trained Gaussian Mixtures [Hegerath et al., 2006] and by Joint Sparse Coding [Yang et al., 2012b]. Below is an extension of our method to work with

arbitrary Q modalities[4]:

$$\begin{aligned} \boldsymbol{\phi} &= \arg\min_{\bar{\boldsymbol{\phi}}} \sum_{q=1}^{Q} \beta^{(q)} \left\| \boldsymbol{x}^{(q)} - \mathcal{M}^{(q)} \bar{\boldsymbol{\phi}} \right\|_2^2 + \alpha \|\bar{\boldsymbol{\phi}}\|_1 \\ &s.\,t.\ \ \bar{\boldsymbol{\phi}} \geq 0 \end{aligned} \tag{7.34}$$

Sparse Coding from [Lee et al., 2007, Yang et al., 2009] is extended in equation (7.34) by combining Q expressions for quantisation loss with the sparsity term. Weights $\beta^{(1)}, ..., \beta^{(Q)}$ determine the impact of features $\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(Q)}$ and dictionaries $\mathcal{M}^{(1)}, ..., \mathcal{M}^{(Q)}$ in this multi-modal trade-off. One can also impose $\beta^{(1)}+...+\beta^{(Q)}=1$. Equation (7.34) is further rewritten to reduce this problem to ordinary SC:

$$\begin{aligned} \boldsymbol{\phi} &= \arg\min_{\bar{\boldsymbol{\phi}}} \sum_{q=1}^{Q} \left\| \sqrt{\beta^{(q)}} \boldsymbol{x}^{(q)} - \sqrt{\beta^{(q)}} \mathcal{M}^{(q)} \bar{\boldsymbol{\phi}} \right\|_2^2 + \alpha \|\bar{\boldsymbol{\phi}}\|_1 \\ &s.\,t.\ \ \bar{\boldsymbol{\phi}} \geq 0 \end{aligned} \tag{7.35}$$

Vector $\boldsymbol{x}$ and dictionary $\mathcal{M}$ for ordinary SC can be formed by concatenation across Q modalities:

$$\boldsymbol{x} = \begin{bmatrix} \sqrt{\beta^{(1)}} \boldsymbol{x}^{(1)} \\ \vdots \\ \sqrt{\beta^{(Q)}} \boldsymbol{x}^{(Q)} \end{bmatrix}, \ \mathcal{M} = \begin{bmatrix} \sqrt{\beta^{(1)}} \mathcal{M}^{(1)} \\ \vdots \\ \sqrt{\beta^{(Q)}} \mathcal{M}^{(Q)} \end{bmatrix} \tag{7.36}$$

Spatial Coordinate Coding described in section 5.2 is used in experiments instead of Spatial Pyramid Matching unless stated otherwise. The descriptor vectors $\boldsymbol{x}$ are augmented with their spatial positions $\boldsymbol{x}^s = [c^x/w, c^y/h]^T$ that are normalised by the image width and height. Thus $\boldsymbol{x} := [\sqrt{\beta^s}\boldsymbol{x}^{sT}, \sqrt{1-\beta^s}\boldsymbol{x}^T]^T$. The trade-off between the visual appearance and spatial bias is balanced by β^s (determined by cross-validation).

Opponent SIFT is comprised of two modalities. The orientations of gradients are extracted from the luminance and chromaticity maps to form two vectors that are normalised and concatenated into the final descriptor. We consider such a descriptor to be based on the augmentation of $\boldsymbol{x}$ with spatial and colour terms $\boldsymbol{x}^s$ and $\boldsymbol{x}^c$ being balanced by β^s and β^c. This results in $\boldsymbol{x} := [\sqrt{\beta^s}\boldsymbol{x}^{sT}, \sqrt{1-\beta^s-\beta^c}\boldsymbol{x}^T, \sqrt{\beta^c}\boldsymbol{x}^{cT}]^T$. This fusion is used only for comparisons with the extension given in section 7.3.3.

[4]Note that symbol Q denoted the number of Pyramid Matching partitions in chapter 6. As Pyramid matching is used sporadically in this chapter, we reuse Q in the context of Q modalities to code.

7.3.2 Late Fusion in Bag-of-Words

Another extremely popular approach to fusing multiple modalities is performed on the kernel level [Nilsback and Zisserman, 2008b,a, Yan et al., 2010, Tahir et al., 2010, Binder et al., 2011]. Typically, multiple modalities are coded and pooled and various kernels are formed to become linearly combined:

$$Ker_{ij} = \sum_{q=1}^{Q} \beta^{(q)} Ker_{ij}^{(q)} \tag{7.37}$$

Weights $\beta^{(1)}, ..., \beta^{(Q)}$ determine the impact of kernels $Ker^{(1)}, ..., Ker^{(Q)}$. One can further impose that $\beta^{(1)}+...+\beta^{(Q)}=1$. There are various approaches to learning weights. However, given a small number of modalities, these weights can be easily found by cross-validation and result in performance on a par with MKL [Yan et al., 2010, Tahir et al., 2010]. This fusion is used only for comparisons to the fusion in section 7.3.3.

7.3.3 Linearisation of Minor Polynomial Kernel for Bi- and Multi-modal Codes

The proposed BoW with Higher-order Occurrence Pooling for bi- and multi-modal codes can be derived in the following four steps: i) defining a kernel function referred to as Minor Kernel on Q pairs of mid-level features, one pair $\left(\boldsymbol{\phi}_n^{(q)}, \bar{\boldsymbol{\phi}}_{\bar{n}}^{(q)}\right)$ per modality $q=1, ..., Q$, ii) summing over pairs of mid-level features formed from two images, iii) normalising with respect to the feature count, iv) normalising the final kernel. First, we define Minor Polynomial Kernel:

$$ker\left(\left\{\left(\boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)}\right)\right\}_{q=1}^{Q}\right) = \left(\sum_{q=1}^{Q} \beta^{(q)} \boldsymbol{\phi}^{(q)^T} \bar{\boldsymbol{\phi}}^{(q)} + \lambda\right)^r \tag{7.38}$$

We chose $\lambda = 0$, while $\beta^{(1)}, ..., \beta^{(Q)}$ are weights determining the impact of modalities, and $r \geq 1$ denotes the polynomial degree (the order of occurrence pooling). One can further impose $\beta^{(1)}+...+\beta^{(Q)}=1$. Equation (7.38) can be rewritten by using the dot product $\left\langle \boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)} \right\rangle$ of a pair of mid-level features:

$$ker\left(\left\{\left(\boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)}\right)\right\}_{q=1}^{Q}\right) = \left(\sum_{q=1}^{Q} \beta^{(q)} \left\langle \boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)} \right\rangle\right)^r \tag{7.39}$$

We assume that $\boldsymbol{\phi}^{(q)}$ and $\bar{\boldsymbol{\phi}}^{(q)}$ are both ℓ_2 norm normalised. Next, we also define a kernel function between two sets of sets of mid-level features $\boldsymbol{\Phi} = \left\{ \left\{ \boldsymbol{\phi}_n^{(q)} \right\}_{n \in \mathcal{N}} \right\}_{q=1}^{Q}$ and $\bar{\boldsymbol{\Phi}} = \left\{ \left\{ \bar{\boldsymbol{\phi}}_{\bar{n}}^{(q)} \right\}_{\bar{n} \in \bar{\mathcal{N}}} \right\}_{q=1}^{Q}$ given descriptor indexes $\mathcal{N}$ and $\bar{\mathcal{N}}$ from two images and given Q modalities:

$$
\begin{aligned}
Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) &= \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|} \sum_{\bar{n} \in \bar{\mathcal{N}}} ker\left(\left\{ \left(\boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)} \right) \right\}_{q=1}^{Q} \right) \\
&= \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|} \sum_{\bar{n} \in \bar{\mathcal{N}}} \left(\sum_{q=1}^{Q} \beta^{(q)} \left\langle \boldsymbol{\phi}^{(q)}, \bar{\boldsymbol{\phi}}^{(q)} \right\rangle \right)^r \\
&= \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \frac{1}{|\bar{\mathcal{N}}|} \sum_{\bar{n} \in \bar{\mathcal{N}}} \left(\sum_{q=1}^{Q} \beta^{(q)} \sum_{k=1}^{K} \phi_{kn}^{(q)} \bar{\phi}_{k\bar{n}}^{(q)} \right)^r
\end{aligned}
\tag{7.40}
$$

Bi-modal Second-order Occurrence Pooling is first derived by linearising the above kernel by setting parameters $Q = 2$ (two coders) and $r = 2$ (second-order). We denote $\beta^{(1)} = \beta$ and $\beta^{(2)} = 1 - \beta$. Thus, Minor Polynomial Kernel from equation (7.39) that appears on the right side of equation (7.40) can be rewritten as:

$$
\left(\beta \sum_{k=1}^{K} \phi_{kn}^{(1)} \bar{\phi}_{k\bar{n}}^{(1)} + (1-\beta) \sum_{k=1}^{K} \phi_{kn}^{(2)} \bar{\phi}_{k\bar{n}}^{(2)} \right)^2 \tag{7.41}
$$

$$
\begin{aligned}
&= \beta^2 \left(\sum_{k=1}^{K} \phi_{kn}^{(1)} \bar{\phi}_{k\bar{n}}^{(1)} \right)^2 + (1-\beta)^2 \left(\sum_{k=1}^{K} \phi_{kn}^{(2)} \bar{\phi}_{k\bar{n}}^{(2)} \right)^2 \\
&\quad + 2\beta(1-\beta) \left(\sum_{k=1}^{K} \phi_{kn}^{(1)} \bar{\phi}_{k\bar{n}}^{(1)} \right) \left(\sum_{k=1}^{K} \phi_{kn}^{(2)} \bar{\phi}_{k\bar{n}}^{(2)} \right)
\end{aligned}
$$

$$
= \beta^2 \left\langle u_{:}^{*}\left(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(1)T} \right), u_{:}^{*}\left(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)T} \right) \right\rangle \tag{7.42}
$$

$$
+ 2\beta(1-\beta) \left\langle u_{:}^{*}\left(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(2)T} \right), u_{:}^{*}\left(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)T} \right) \right\rangle \tag{7.43}
$$

$$
+ (1-\beta)^2 \left\langle u_{:}^{*}\left(\boldsymbol{\phi}_n^{(2)} \boldsymbol{\phi}_n^{(2)T} \right), u_{:}^{*}\left(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)T} \right) \right\rangle \tag{7.44}
$$

Minor Polynomial Kernel in equation (7.41) is linearised for order $r = 2$ with three dot product terms in equations (7.42), (7.43), and (7.44). Substituting Minor Polynomial

Kernel in equation (7.40) by these terms yields:

$$
\begin{aligned}
&Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) = Ker_{ij} \\
&= \beta^2 \Big\langle \underset{n \in \mathcal{N}}{\text{avg}} \left[u_{:}^{*} \left(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(1)T} \right) \right], \underset{\bar{n} \in \bar{\mathcal{N}}}{\text{avg}} \left[u_{:}^{*} \left(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)T} \right) \right] \Big\rangle && (7.45) \\
&+ 2\beta(1-\beta) \Big\langle \underset{n \in \mathcal{N}}{\text{avg}} \left[u_{:}^{*} \left(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(2)T} \right) \right], \underset{\bar{n} \in \bar{\mathcal{N}}}{\text{avg}} \left[u_{:}^{*} \left(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(1)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)T} \right) \right] \Big\rangle && (7.46) \\
&+ (1-\beta)^2 \Big\langle \underset{n \in \mathcal{N}}{\text{avg}} \left[u_{:}^{*} \left(\boldsymbol{\phi}_n^{(2)} \boldsymbol{\phi}_n^{(2)T} \right) \right], \underset{\bar{n} \in \bar{\mathcal{N}}}{\text{avg}} \left[u_{:}^{*} \left(\bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)} \bar{\boldsymbol{\phi}}_{\bar{n}}^{(2)T} \right) \right] \Big\rangle && (7.47)
\end{aligned}
$$

Note that the final kernel for the two coders is comprised of three dot product terms. Equations (7.45) and (7.47) represent simply Second-order Occurrence Pooling for coders $q=1$ and $q=2$. They are identical with the uni-modal coding given by equation (7.22) in section 7.2.1. However, equation (7.46) represents the cross-term that captures co-occurrences between visual words of mid-level features $\phi_{kn}^{(1)}$ and $\phi_{k'n}^{(2)}$ from two coders. The cross-term will be shown later to improve results.

In practice, we work with Second-order Occurrence Pooling and the @n operator, as in section 7.2.2. The earlier defined operator $u_{:}$ is used in place of $u_{:}^{*}$ in equations (7.45) and (7.47) to reject the redundant coefficients from the symmetric self-tensor products. Lastly, the image signatures are the ℓ_2 norm normalised.

Bi-modal Higher-order Occurrence Pooling can be derived from expansion of Minor Polynomial Kernel in equation (7.39) using Binomial theorem:

$$
\Big[\beta a + (1-\beta) b \Big]^r = \sum_{s=0}^{r} \binom{r}{s} \Big[\beta a \Big]^{r-s} \Big[(1-\beta) b \Big]^s \tag{7.48}
$$

Two coders $Q=2$ and order $r \geq 2$ are assumed, and substitutions $a = \langle \boldsymbol{\phi}^{(1)}, \bar{\boldsymbol{\phi}}^{(1)} \rangle$ and $b = \langle \boldsymbol{\phi}^{(2)}, \bar{\boldsymbol{\phi}}^{(2)} \rangle$ are made. The derivations follow the same reasoning as for Bi-modal Second-order Occurrence Pooling. We skip them for clarity and define Bag-of-Words

with Bi-modal Higher-order Occurrence Pooling:

$$\begin{aligned}\boldsymbol{\phi}_n^{(1)} &= f^{(1)}\left(\boldsymbol{x}_n^{(1)}, \mathcal{M}^{(1)}\right) \\ \boldsymbol{\phi}_n^{(2)} &= f^{(2)}\left(\boldsymbol{x}_n^{(2)}, \mathcal{M}^{(2)}\right)\end{aligned}, \quad \forall n \in \mathcal{N} \tag{7.49}$$

$$\boldsymbol{\psi}_n^s = u_{:}\left[\left(\otimes_{r-s}\, \boldsymbol{\phi}_n^{(1)}\right)\left(\otimes_s\, \boldsymbol{\phi}_n^{(2)}\right)\right],\ s = 0, ..., r \tag{7.50}$$

$$\hat{h}_k^s = \binom{r}{s}^{\frac{1}{2}} (1-\beta)^{\frac{s}{2}}\, \beta^{\frac{r-s}{2}}\, g\left(\{\psi_{kn}^s\}_{n\in\mathcal{N}}\right),\ k = 1, ..., K^{(r,s)} \tag{7.51}$$

$$\boldsymbol{h} = \hat{\boldsymbol{h}}/\|\hat{\boldsymbol{h}}\|_2\ ,\ \hat{\boldsymbol{h}} = \left[\hat{\boldsymbol{h}}^{\boldsymbol{0}T}, ..., \hat{\boldsymbol{h}}^{\boldsymbol{r}T}\right]^T \tag{7.52}$$

Figure 7.6 illustrates Bi-modal BoW with Second-order Occurrence Pooling.

Equation (7.49) represents the coding step for two coders $f^{(1)} : \mathbb{R}^{D^{(1)}} \rightarrow \mathbb{R}^{K^{(1)}}$ and $f^{(2)} : \mathbb{R}^{D^{(2)}} \rightarrow \mathbb{R}^{K^{(2)}}$ that embed descriptors $\boldsymbol{x}_n^{(1)} \in \mathbb{R}^{D^{(1)}}$ and $\boldsymbol{x}_n^{(2)} \in \mathbb{R}^{D^{(2)}}$ representing two modalities into the visual vocabulary spaces given by dictionaries $\mathcal{M}^{(1)} \in \mathbb{R}^{D^{(1)} \times K^{(1)}}$ and $\mathcal{M}^{(2)} \in \mathbb{R}^{D^{(2)} \times K^{(2)}}$. This results in two groups of mid-level features $\boldsymbol{\phi}_n^{(1)} \in \mathbb{R}^{K^{(1)}}$ and $\boldsymbol{\phi}_n^{(2)} \in \mathbb{R}^{K^{(2)}}$ given the descriptor indexes $n \in \mathcal{N}$ of image $i \in \mathcal{I}$. Moreover, the coders

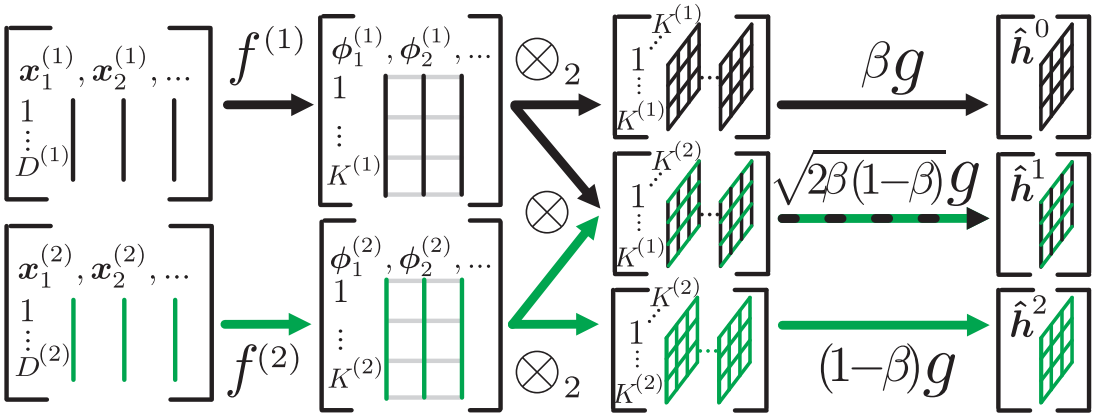


Figure 7.6: Bi-modal Bag-of-Words with Second-order Occurrence Pooling. Two types of local descriptors $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ are extracted from an image and coded by coders $f^{(1)}$ and $f^{(2)}$. Self-tensor product $\otimes_2$ computes co-occurrences of visual words in every mid-level feature $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$, respectively. Moreover, tensor product $\otimes$ captures co-occurrences of visual words between $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$ (cross-term operation). Pooling g aggregates co-occurring visual words. For clarity, the flattening operator $u_{:}$ from equation (7.50) is dropped.

used can be of different types, the descriptor dimensionality $D^{(1)}$ may differ from $D^{(2)}$, and dictionary sizes $K^{(1)}$ and $K^{(2)}$ may differ.

Equation (7.50) represents the joint occurrence of visual words in $\boldsymbol{\phi}_n^{(1)}$ or $\boldsymbol{\phi}_n^{(2)}$, or the cross-modal joint occurrence of visual words per mid-level pair $(\boldsymbol{\phi}_n^{(1)}, \boldsymbol{\phi}_n^{(2)})$, depending on k and s. It results from an expansion of Minor Polynomial Kernel in equation (7.39) according to Binomial theorem. A similar expansion was performed in equations (7.41-7.44) for $r=2$. However, we moved weight β inside the dot product and conveniently appended them to the pooling operator in equation (7.51). Thus, only vectors $\boldsymbol{\psi}_n^s$ that would appear inside the dot product expressions are given. Furthermore, equation (7.50) uses the previously defined operator $u_:$ rather than $u_:^*$ to reject the redundant coefficients from the symmetric self-tensor products. This operator is applied here to symmetries that occur in self-tensors $\otimes_{r-s}\boldsymbol{\phi}_n^{(1)}$ and $\otimes_s\boldsymbol{\phi}_n^{(2)}$ if $r-s \geq 2$ or $s \geq 2$. The dimensionality of $\boldsymbol{\psi}_n^s$ after rejecting repeated coefficients and flattening is $K^{(r,s)} = K^{(r-s)}K^{(s)} = \binom{K+r-s-1}{r-s}\binom{K+s-1}{s}$.

Equation (7.51) is the pooling step that aggregates the joint occurrences or the cross-modal joint occurrences of visual words. Function $g : \mathbb{R}^{|\mathcal{N}|} \rightarrow \mathbb{R}$ takes k^{th} the joint occurrence (or the cross-modal joint occurrence) coefficients ψ_{kn}^s for all $n \in \mathcal{N}$ given image i to produce a k^{th} coefficient in vector $\hat{\boldsymbol{h}}^s \in \mathbb{R}^{K^{(r,s)}}$. The weighting factor in front of g results from Binomial expansion. We mainly use the $@n$ operator for this step.

Equation (7.52) concatenates various joint occurrence statistics and also performs the ℓ_2 norm normalisation.

Bi-modal Second-order Occurrence Pooling in equations (7.45), (7.46), and (7.47) can also be readily derived from Bi-modal Higher-order Occurrence Pooling. If $r=2$, then equation (7.50) results in three terms:

$$\boldsymbol{\psi}_n^0 = u_:^*\left(\boldsymbol{\phi}_n^{(1)}\boldsymbol{\phi}_n^{(1)T}\right), \hat{h}_k^0 = \beta \operatorname{avg}\left(\left\{\psi_{kn}^0\right\}_{n\in\mathcal{N}}\right) \tag{7.53}$$

$$\boldsymbol{\psi}_n^1 = u_:^*\left(\boldsymbol{\phi}_n^{(1)}\boldsymbol{\phi}_n^{(2)T}\right), \hat{h}_k^1 = \sqrt{2\beta(1-\beta)}\operatorname{avg}\left(\left\{\psi_{kn}^1\right\}_{n\in\mathcal{N}}\right) \tag{7.54}$$

$$\boldsymbol{\psi}_n^2 = u_:^*\left(\boldsymbol{\phi}_n^{(2)}\boldsymbol{\phi}_n^{(2)T}\right), \hat{h}_k^2 = (1-\beta) \operatorname{avg}\left(\left\{\psi_{kn}^2\right\}_{n\in\mathcal{N}}\right) \tag{7.55}$$

Employing Average pooling for the pooling step in equation (7.51) is done by replacing g with avg. Pooling over $\boldsymbol{\psi}_n^0$, $\boldsymbol{\psi}_n^1$, and $\boldsymbol{\psi}_n^2$ given in equations (7.53), (7.54), and (7.55)

results in $\hat{\boldsymbol{h}}^0$, $\hat{\boldsymbol{h}}^1$, and $\hat{\boldsymbol{h}}^2$ per image. Forming three kernels $\langle \hat{\boldsymbol{h}}_i^0, \hat{\boldsymbol{h}}_j^0 \rangle$, $\langle \hat{\boldsymbol{h}}_i^1, \hat{\boldsymbol{h}}_j^1 \rangle$, and $\langle \hat{\boldsymbol{h}}_i^2, \hat{\boldsymbol{h}}_j^2 \rangle$ given images i and j and adding these kernels is equivalent to operations in equations (7.45), (7.46), and (7.47).

Multi-modal Higher-order Occurrence Pooling can be readily derived by expanding Minor Polynomial Kernel in equation (7.39) using Multinomial theorem. Furthermore, this type of fusing multiple modalities can be realised simply by concatenating the mid-level features of index n from Q coders:

$$\boldsymbol{\phi}_n = \left[\sqrt{\beta^{(1)}} \boldsymbol{\phi}_n^{(1)T}, \sqrt{\beta^{(2)}} \boldsymbol{\phi}_n^{(2)T}, ..., \sqrt{\beta^{(Q)}} \boldsymbol{\phi}_n^{(Q)T} \right]^T \tag{7.56}$$

Such formed super mid-level features $\boldsymbol{\phi}_n$ can be used to form a tensor according to equation 7.15. This formulation is compatible with the proposed above Bi- and Multi-modal Second- and Higher-order Occurrence Pooling.

7.3.4 Special Cases of Bi-modal Second-order Occurrence Pooling: Pyramid Matching Techniques

Spatial Pyramid Matching (SPM) from [Lazebnik et al., 2006, Yang et al., 2009] is demonstrated now as special cases of Bi-modal Second-order Occurrence Pooling. We employ two coders such that f is SC, LLC, LcSA, or other coding, and the second coder produces a binary vector with assignments of descriptors to spatial partitions:

$$\begin{aligned} \boldsymbol{\phi}_n^{(1)} &= f\left(\boldsymbol{x}_n, \mathcal{M}\right) \\ \boldsymbol{\phi}_n^{(2)} &= \left[\oplus_{t=1}^{\overline{T}} \oplus_{zx=0}^{Z_t-1} \oplus_{zy=0}^{\overline{Z}_t-1} \mathbb{1}\left(\left\lfloor \frac{Z_t c_n^x}{w} \right\rfloor = z_x \right) \mathbb{1}\left(\left\lfloor \frac{\overline{Z}_t c_n^y}{h} \right\rfloor = z_y \right) \right]^T \end{aligned} \tag{7.57}$$

Equation (7.57) uses the operator $\oplus_{t=1}^{\overline{T}}$ denoting concatenation over $\overline{T}$ levels of spatial quantisation. Operators $\oplus_{zx=0}^{Z_t-1}$ and $\oplus_{zy=0}^{\overline{Z}_t-1}$ concatenate binary values over vertical and horizontal partitions $z_x = 0, ..., Z_t - 1$ and $z_y = 0, ..., \overline{Z}_t - 1$, where vectors $\boldsymbol{Z}$ and $\overline{\boldsymbol{Z}}$ define the numbers of splits for each pyramid level $t = 1, ..., \overline{T}$. Binary indicator $\mathbb{1}(z_l = z_r)$ returns 1 if $z_l = z_r$, 0 otherwise. Next, $0 \leq c_n^x < w$ and $0 \leq c_n^y < h$ are the spatial coordinates of descriptor $\boldsymbol{x}_n$, w and h are the image width and height, and $\lfloor . \rfloor$ is the floor operator.

SPM (*e.g.* variant from [Yang et al., 2009]) can be obtained by simply applying Bi-modal Second-order Occurrence Pooling, extracting the cross-modal joint occurrence of visual words that form $\boldsymbol{\psi}_n^1$, and suppressing the joint occurrence of visual words in $\boldsymbol{\psi}_n^0$ and $\boldsymbol{\psi}_n^2$:

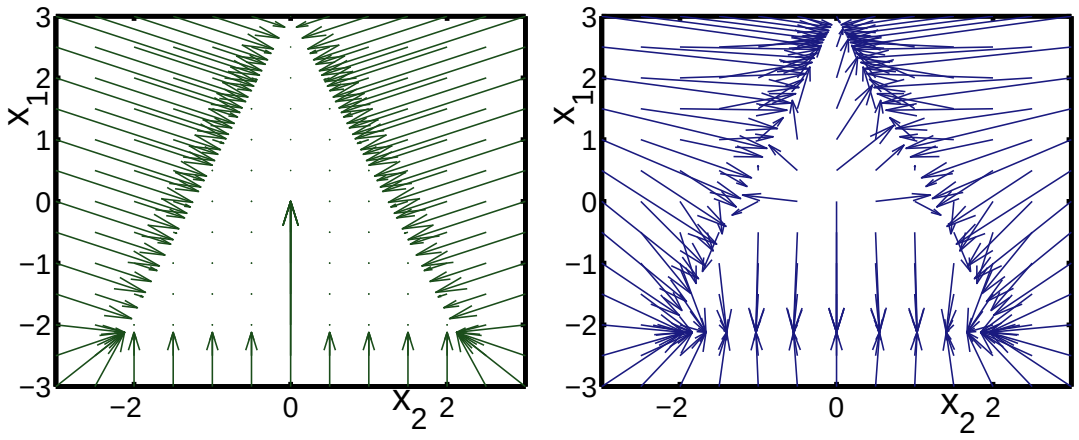
$$\boldsymbol{\psi}_n^0 = [\,], \boldsymbol{\psi}_n^1 = u_{:}^{*}\left(\boldsymbol{\phi}_n^{(1)} \boldsymbol{\phi}_n^{(2)T}\right), \boldsymbol{\psi}_n^2 = [\,] \tag{7.58}$$

The parameters for SPM with 1×1, 3×1, 1×3, and 2×2 spatial splits are $\overline{T}$=4, $\boldsymbol{Z}$=[1 3 1 2]^T and $\overline{\boldsymbol{Z}}$=[1 1 3 2]^T. SPM gathers second-order statistics by quantifying co-occurrences between visual words in the mid-level features and spatial locations that are quantised at several levels of quantisation. Thus, SPM enhances the visual vocabulary with a spatial vocabulary: similar visual appearances can take various meanings based on their spatial locations. A similar mechanism is explained in section 7.2.3. Moreover, we stress that Bi-modal Second-order Occurrence Pooling actually results in three terms $\boldsymbol{\psi}_n^0$, $\boldsymbol{\psi}_n^1$, and $\boldsymbol{\psi}_n^2$. Therefore, it is worthy to evaluate such an SPM model.

By analogy to SPM, DoPM proposed in section 5.3 can be obtained by re-defining the coder in equation (7.58) to exploit orientations of dominant edges from the local descriptors in place of spatial coordinates. BoW schemes like BossaNova from [Avila et al., 2012] can be also derived by employing: i) the descriptor assignment to l-nearest k-means clusters as the first coder, ii) the descriptor assignment to radial zones defined over k-means clusters as the second coder.

7.3.5 Residual Descriptor

We now present the Residual Descriptor (RD) that is used along with a chosen coder (*e.g.* SC, LLC, or LcSA) to address its quantisation loss. RD is not related to bi-modal fusion, however, we illustrate an interesting property of Bi-modal Second-order Occurrence Pooling with its means. To achieve good performance, SC and LLC optimise a trade-off between a quantisation loss (defined below) and an explicitly chosen regularisation penalty, *e.g.* sparsity as in equation (6.6) or locality as in equation (6.8). The quality of quantisation in these mappings is measured in accordance with the theory of Linear Coordinate Coding [Yu et al., 2009] already presented in section 4.3 of chapter 4. The linear approximation of descriptor $\boldsymbol{x}$ given visual dictionary $\mathcal{M}$ and coder f



(*a*) SC, α=1 (*b*) LcSA, σ^2=4, l=2

Figure 7.7: Illustration of Residual Descriptors. Flow of the descriptors from their original positions $\boldsymbol{x}$ denoted by the grid points to the corresponding reconstructed positions $\hat{\boldsymbol{x}}$ pointed to by the arrows. (*a*) SC: optimal reconstruction within the triangular region. (*b*) LcSA: case of limited reconstruction due to low $l=2$.

that produces mid-level feature $\boldsymbol{\phi}$ is $\hat{\boldsymbol{x}} = \mathcal{M} f(\boldsymbol{x}) = \mathcal{M}\boldsymbol{\phi}$. The quantisation loss a.k.a quantisation error is defined as the residual sum of squares:

$$\xi^2(\boldsymbol{x}) = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2 \tag{7.59}$$

However, $\xi^2(\boldsymbol{x})$ quantifies only the magnitude of such an error. Therefore, we define a Residual Descriptor vector that can capture also the phase:

$$\boldsymbol{\xi}(\boldsymbol{x}) = \boldsymbol{x} - \hat{\boldsymbol{x}} \tag{7.60}$$

Residual Descriptors have been already illustrated in figure 6.3 from chapter 6. For convenience, figure 7.7 zooms at SC and LcSA related plots. Having coded descriptors $\boldsymbol{x} = [x_1, x_2]^T \in \langle -3; 3\rangle^2$ with three atoms $\boldsymbol{m}_1 = [0, 3]^T$, $\boldsymbol{m}_2 \approx [-2, -2]^T$, and $\boldsymbol{m}_3 \approx [2, -2]^T$ by SC and LcSA coders, the obtained codes $\boldsymbol{\phi}$ are projected back to the descriptor space: $\hat{\boldsymbol{x}} = \mathcal{M}\boldsymbol{\phi}$. The resulting quantisation artifacts are visualised as displacements between each descriptor $\boldsymbol{x}$ and its approximation $\hat{\boldsymbol{x}}$. Plots (a, b) illustrate the SC and LcSA cases with low and large quantisation errors, respectively.

The displacements in figure 6.3 are shown with respect to descriptors $\boldsymbol{x}$. However, encoding the magnitude and orientation of the quantisation error given equation (7.60) does not indicate which descriptors are the source of errors. Hence, we propose to use Bi-modal Second-order Occurrence Pooling framework to combine both mid-level features $\boldsymbol{\phi}$ and vectors $\boldsymbol{\xi}$:

$$\begin{aligned} \boldsymbol{\phi}_n^{(1)} &= f\left(\boldsymbol{x}_n, \mathcal{M}\right) \\ \boldsymbol{\phi}_n^{(2)} &= \boldsymbol{x}_n - \mathcal{M}\boldsymbol{\phi}_n^{(1)} \end{aligned} \tag{7.61}$$

In this formulation, the cross-term captures co-occurrences between visual words of mid-level feature $\boldsymbol{\phi}^{(1)}$ of descriptor $\boldsymbol{x}$ and directions of the corresponding residual error $\boldsymbol{\xi}$. This associates the error with the descriptor and helps us correct for the coding artifacts. We demonstrate later that the cross-term resulting from this formulation is very informative. Lastly, a somewhat related approach to the residual error was proposed in [Zhang et al., 2012]. The quantisation loss is computed with respect to one atom at a time. The resulting code is appended to a corresponding mid-level feature.

7.4 Experimental Section

The proposed Second- and Higher-order Occurrence Pooling methods are evaluated on the PascalVOC07 [Everingham et al., 2007], Caltech101 [Fei-fei et al., 2004], Flower102 [Nilsback and Zisserman, 2008a], and ImageCLEF11 [Nowak et al., 2011] datasets.

7.4.1 Experimental Arrangements and Datasets

The PascalVOC07 [Everingham et al., 2007] set consists of 20 classes of objects of varied nature, *e.g.* *human*, *cat*, *chair*, *train*, *bottle*. This is a challenging collection of images with objects that appear at variable scales and orientations, often in difficult visual contexts and backgrounds, being frequently partially occluded. We use this set for the whole spectrum of proposed experiments and use the training, validation, and testing splits as provided. The Caltech101 [Fei-fei et al., 2004] set consists of 101 classes represented by objects which are aligned to the centres of images as well as a separate background class. The evaluations are performed with 15 and 30 training

Dataset	Splits no.	Training+validation samples	Test samples	Total images	Dict. size	Descr. type	Dims. (grey+col.)
PascalVOC07	1x	2501+2510=5011	4952	9963	100-1600	Opp. SIFT	128D+ 144D
Caltech101	10x	12+3=15/24+6=30	rest	9144	300-800	SIFT	128D
Flower102	1x	1020+1020=2040	6149	8189	300-1600	Opp. SIFT	128D+ 144D
ImageCLEF11		6K+2K=8K (+8K flip)	10K	18K (+8K)	800		

	Descr. interval	Radii (px)	Descr. per img.	Coding	Spatial/other schemes	Order	Kernel types	Classifier used
PascalVOC07	4,6,8,10, 12,14,16	12,16,24,32, 40,48,56	19420	SC/LLC/ LcSA	none/SCC/ SPM*/DoPM*	1*,2,3	linear	multilabel
Caltech101	4,6,8,10	16,24,32,40	5200	SC	SCC/SPM*	1*,2		multiclass
Flower102	6,9,12,15		14688		SCC/DoPM*	1*,2		
ImageCLEF11	4,6,8,10, 12,14,16	12,16,24,32, 40,48,56	19642		SCC	2	linear/ χ^2_{RBF}	multilabel

(*) used in comparisons only

Table 7.1: Summary of the datasets, descriptor parameters, and experimental details.

images per class. The Flower102 [Nilsback and Zisserman, 2008a] set of 102 flower classes was used for further evaluations. A single split into the training and testing sets is supplied for this corpus. ImageCLEF11 Photo Annotation [Nowak et al., 2011] is a challenging collection of images represented by 99 concepts of a varied nature, including complex topics, *e.g. party life*, *funny*, *work*, *birthday*. Unlike sets of objects, this challenge aims at annotation labels that correspond to human-like understanding of a scene. ImageCLEF11 is a subset of MIRFLICKR with vastly improved annotations which enables better classification [Huiskes and Lew, 2008, Huiskes et al., 2010]. Only the visual annotation was used for this dataset. To best use the available images in ImageCLEF11, the training set was doubled by left-right flipping training images [Chatfield et al., 2011]. Table 7.1 presents the experimental parameters for all datasets.

Dictionaries. Online Dictionary Learning was used to train dictionaries for Sparse Coding [Mairal et al., 2010]. Dictionary learning proposed for Approximate Locality-constrained Linear Coding [Wang et al., 2010] was used for this coder. Furthermore, we adapted such a method to work with Approximate Locality-constrained Soft Assignment as it outperformed LcSA with dictionaries formed by k-means. Size-wise, we used between $4K$ to $40K$ for First-, 300 to 1600 for Second-, and 100 to 200 for Third-order Occurrence Pooling. Fisher Vector Encoding [Perronnin and Dance, 2007, Perronnin et al., 2010, Sánchez et al., 2012] and Vector of Locally Aggregated Tensors [Negrel

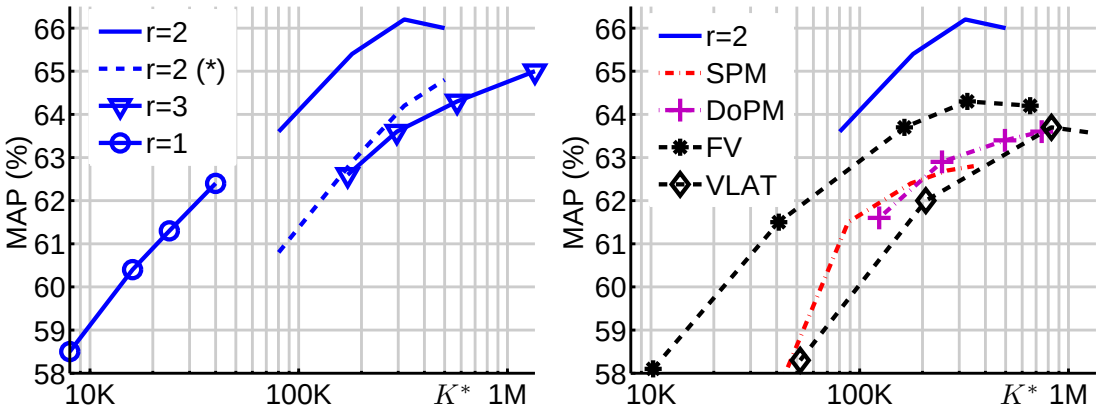
et al., 2012] were used in comparisons, GMM and k-means dictionaries with 64 to 4096 and 64 to 512 atoms were employed, respectively.

Descriptors. Opponent SIFT was extracted on dense grids. The grey scale components ($128D$) were used for uni-modal BoW. The colour components ($144D$) were additionally used for bi-modal BoW. No PCA was applied except for FK and VLAT ($80D$ for the grey and $120D$ for the grey and opponent components).

Dataset bias. Spatial relations in images were exploited mainly by Spatial Coordinate Coding described in section 5.2 and explained in the context of this work in 7.3.1. SPM and DoPM were additionally used to: i) obtain comparative results on the standard BoW (first-order), ii) evaluate the proposed special cases of SPM and DoPM given in section 7.3.4. SPM used 3 levels of coarseness with 1×1, 1×13, 3×11, and 2×12 grids on PascalVOC07, and 4 levels with 1×11, 2×12, 3×13, and 4×14 grids on Caltech101. DoPM was used to exploit dominant edge bias given 5 levels of coarseness with 1, 3, 6, 9, and 12 grids on PascalVOC07, and 3 levels with 1, 2, and 3 grids on Flower102. Comparisons on the standard BoW (first-order) employed either SCC, SPM, or DoPM. By default, all experiments on DoPM used the descriptor coordinates appended at the descriptor level (SCC). Applying SPM directly to Second-order Occurrence Pooling performed worse than SCC, produced extremely large signatures, thus it is rarely reported on. Similar findings were presented in [Sánchez et al., 2012] for FK combined with SCC rather than SPM. Thus, we combined FK and VLAT with SCC.

Coding and Pooling. We used SC for the most of experiments except for additional demonstrations of Second-order Occurrence Pooling with LLC and LcSA. The pooling operator $@n$ was used throughout experiments, however, a brief comparison on Max-pooling, MaxExp, and Power Normalisation is provided. FK and VLAT were combined with PN only as other operators are not directly applicable here. In all cases, we determined the coding and pooling parameters by cross-validation. Moreover, all comparative results on the standard BoW (first-order) used SC with the $@n$ operator.

Kernels. Linear kernels $Ker_{ij} = (\boldsymbol{h}_i)^T \cdot \boldsymbol{h}_j$ were used, where $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^K$ are image signatures for $i, j \in \mathcal{I}$. χ^2 merged with RBF (χ^2_{RBF}) defined as $Ker_{ij} = exp\,[-\rho^2 \sum_k (h_{ki} - h_{kj})^2/(h_{ki} + h_{kj})]$ was used additionally on ImageCLEF11, $1/\rho$ is the RBF radius.



(*a*) VOC07, $r=1,2,3$ (*b*) VOC07, $r=2$ vs other methods

Figure 7.8: Performance of Higher-order Occurrence Pooling compared to various approaches on the PascalVOC07 set. Results were plotted as functions of the signature length K^*. (*a*) First-, Second-, and Third-order Occurrence Pooling $r=1,2,3$ with Spatial Coordinate Coding. Asterisk (*) denotes the case of order $r=2$ without any spatial information. (*b*) The case of order $r=2$ compared to SPM and DoPM ($r=1$). Furthermore, results on FK and VLAT were also plotted.

Classifiers. Multi-label KDA from [Tahir et al., 2009] was applied to PascalVOC07 and ImageCLEF11, as it was previously found to be a robust performer on these sets [Tahir et al., 2009, 2010]. The MAP measure is used to report the performance on these sets. Multi-class KDA from [Tahir et al., 2009] was applied to both Clatech101 and Flower102 to process the image signatures. Mean Accuracy is the reported performance measure.

7.4.2 Evaluating Uni-modal Bag-of-Words for First-, Second-, and Third-order Occurrence Pooling

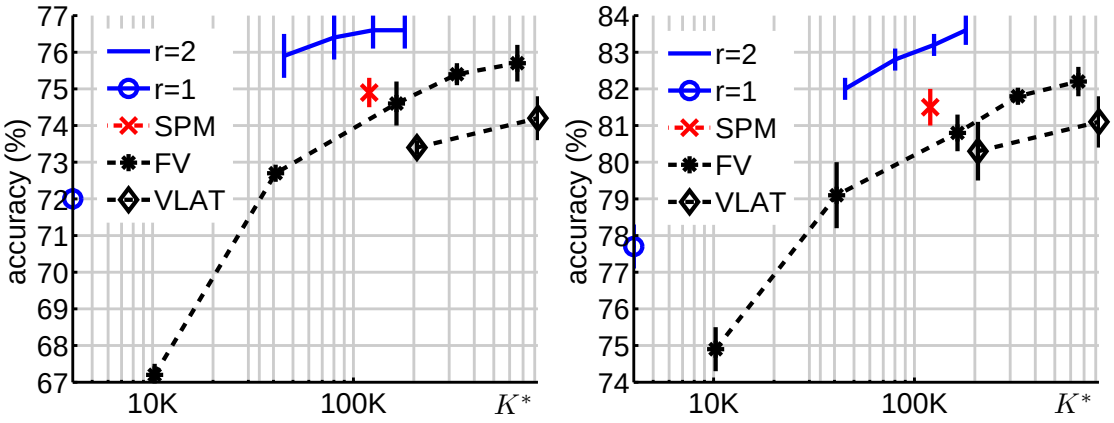
This section presents how BoW described in section 7.2 performed in a practical classification scenario given order $r=1$, 2, and 3, and the grey scale SIFT. Note that $r=1$ renders BoW from section 7.2 to be equivalent to the standard BoW in section 7.1.1.

Figure 7.8 (a) compares the classification performance of the proposed method for various orders r on the PascalVOC07 set (SCC is used). Second-order Occurrence

Pooling is shown to outperform the first- and third-order cases. It attains 65.4%, 66.2%, and 66.0% MAP for $K = 600$, 800, and 1000 dictionary atoms that result in the signature lengths $K^* = 180300$, 320400, and 500500, respectively. Next, First-order Occurrence Pooling scores respectable 62.4% MAP for $K = K^* = 40000$ atoms (this is also the signature length). However, the coding step is computationally prohibitive for large visual dictionaries. It takes 815 and 1.5 seconds to code 1000 descriptors on a single 2.3GHz AMD Opteron core given $K = 40000$ and $K = 800$ atoms, respectively. Third-order Occurrence Pooling yields 65% MAP for $K = 200$ atoms resulting in the signature length $K^* = 1353400$. Our experiments suggest that the second-order case yields the highest results and provides an attractive trade-off between the tractability of coding and the signature lengths. Finally, Second-order Occurrence Pooling without any spatial information attains 64.8% MAP for $K = 1000$ atoms. This demonstrates the benefit of SCC.

Figure 7.8 (b) compares Second-order Occurrence Pooling ($r = 2$, SCC is used) to the standard BoW ($r = 1$) combined with SPM and DoPM, respectively. FK and VLAT combined with SCC are also evaluated. BoW ($r = 1$) with SPM attains 62.8% MAP for $K = 32000$ atoms and results in the signature length $K^* = 352000$. BoW ($r = 1$) with DoPM yields 63.6% MAP and outperforms SPM by 0.8% for $K = 24000$ atoms and the signature length $K^* = 744000$. This is comparable to VLAT that attains 63.7% MAP for the signature length $K^* = 829440$. Lastly, FK yields 64.3% MAP given the signature length $K^* = 327680$. Thus, Second-order Occurrence Pooling outperforms FK by 1.9% MAP for the comparable signature length.

Not included in the plots, Second-order Pooling with SPM applied to raw SIFT as proposed in [Carreira et al., 2012] yields 54.2% MAP only. In this approach, the coding step is bypassed. The results suggest that applying the coding step to learn the data manifold is vital to obtain good results.

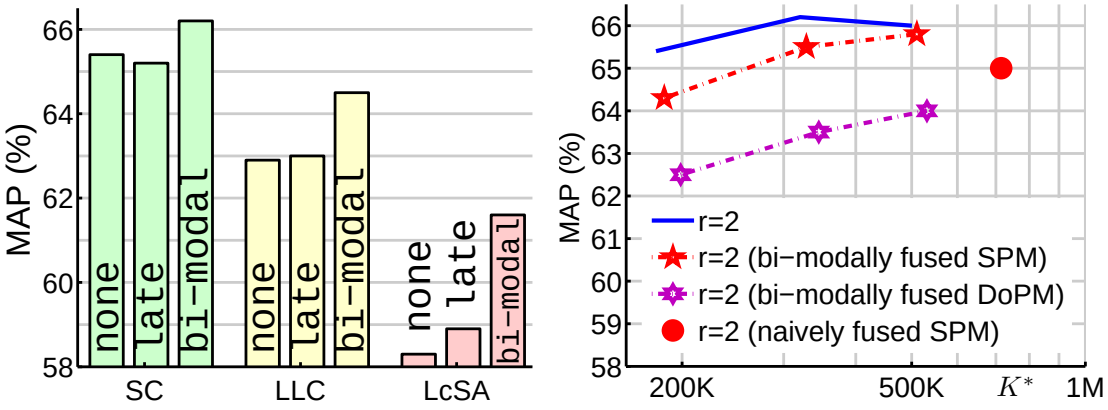


(*a*) Caltech101 (15 samples) (*b*) 30 samples/class

Figure 7.9: Performance of Second-order Occurrence Pooling ($r = 2$) compared to various approaches on the Caltech101 set. Results were plotted as functions of the signature length K^*. Standard BoW of order $r = 1$ with SCC ($r = 1$), BoW with SPM (*SPM*), FK, and VLAT were evaluated on (*a*) 15, and (*b*) 30 training images per class.

Figure 7.9 (a) provides evaluations on 15 training images per class. BoW ($r = 1$) with SCC yields $72 \pm 0.3\%$ accuracy for $K = K^* = 4000$ atoms (this is also the signature length). This offers very compact signatures and a good performance. BoW ($r = 1$) with SPM yields $74.9 \pm 0.4\%$ accuracy for $K = 4000$ atoms and the signature length $K^* = 120000$. This represents a slight improvement over FK that yields $74.6 \pm 0.6\%$ accuracy given the signature length $K^* = 163840$. Lastly, Second-order Occurrence Pooling yields $76.6 \pm 0.5\%$ given $K = 500$ atoms and the signature length $K^* = 125250$. This is a 2% improvement over FK given the comparable signature lengths. FK and VLAT yield $75.7 \pm 0.5\%$ and $74.2 \pm 0.6\%$ accuracy at best.

Figure 7.9 (b) provides evaluations given 30 training images per class. The comparison arrangements remain identical to those presented above. Second-order Occurrence Pooling scores $83.6 \pm 0.4\%$ accuracy given $K = 600$ atoms and the signature length $K^* = 180300$. This is a 2.8% improvement over FK that scores $80.8 \pm 0.5\%$ accuracy for the comparable signature length $K^* = 163840$. BoW ($r = 1$) with SPM yields $81.5 \pm 0.4\%$ accuracy for $K = 4000$ atoms and the signature length $K^* = 120000$. This also represents a small gain of 0.7% over FK. BoW ($r = 1$) with SCC yields $77.7 \pm 0.6\%$ accuracy. FK and VLAT yield $82.2 \pm 0.4\%$ and $81.1 \pm 0.7\%$ accuracy at best.



(*a*) VOC07, Residual Descriptor (*b*) VOC07, fusing SPM

Figure 7.10: Evaluation of Bi-modal Second-order Occurrence Pooling (PascalVOC07). (*a*) Bars (*none*) show results for SC, LLC, and LcSA coders ($r=2$, 600 atoms). Residual Descriptors from section 7.3.5 were fused by the late fusion (*late*) from section 7.3.2 (note little improvement). A larger gain is shown for Bi-modal Second-order Occurrence Pooling (*bi-modal*). (*b*) Special case SPM and DoPM proposed in section 7.3.4 were fused by Bi-modal Second-order Occurrence Pooling (*bi-modally*). SPM applied directly to the mid-level features (*naively*) is also evaluated for $r=2$.

7.4.3 Evaluations of SC, LLC, and LcSA given Uni-modal Second-order Occurrence Pooling

The coding step is now evaluated and demonstrated to have a significant impact on the performance of Second-order Occurrence Pooling. Extensive evaluations for the standard BoW ($r=1$) are provided in chapter 6.

Figure 7.10 (a) demonstrates results on SC, LLC, and LcSA, all obtained on the PascalVOC07 set for $K=600$ dictionary atoms that resulted in the signature lengths $K^*=180300$. Bars (*none*) show that SC yields 65.4%, LLC 62.9%, and LcSA 58.3% MAP. This is in agreement with the observation that the lower the quantisation loss of a coder is, the better the classification results are. We evaluated ξ^2 according to equation (7.59) for a subset of descriptors, summed over the individual ξ^2 for each descriptor, and observed that $\xi^2_{SC} < \xi^2_{LLC} < \xi^2_{LcSA}$. In the next section, we will demonstrate that Residual Descriptor from section 7.3.5 can exploit these quantisation effects.

Finally, we note that the gap in performance between SC and LcSA is 7.1% MAP. We expect that the worse the quantisation properties of a coder are, the more distorted the joint occurrences of visual words on the mid-level feature level become. The gap between SC and LcSA is much smaller for the standard BoW ($r=1$) with SPM, as demonstrated earlier in section 6.4.3.

7.4.4 Evaluations of Bi-modal Bag-of-Words for Second-order Occurrence Pooling

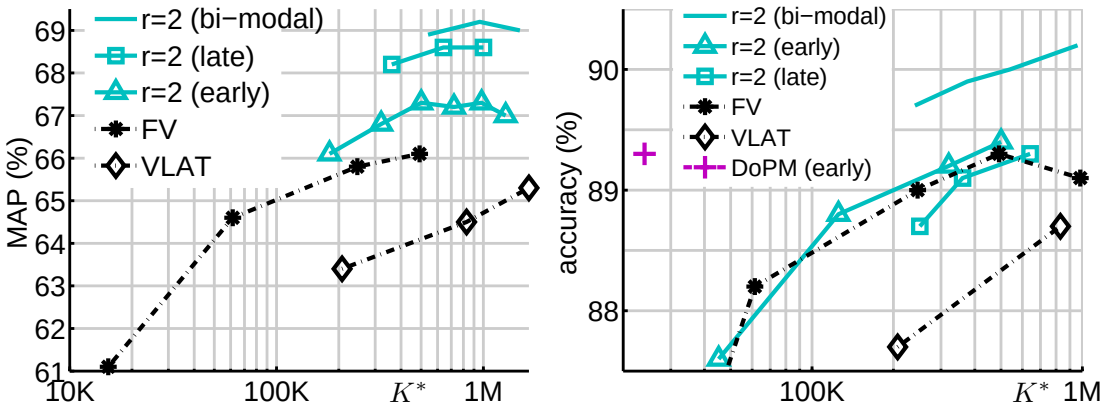
This section presents the classification performance for BoW given order $r=2$ described in section 7.3 and illustrated in figure 7.6. The modalities to fuse are: i) the grey scale SIFT and Residual Descriptor proposed in section 7.3.5, ii) the grey scale SIFT and special case SPM and DoPM, respectively, as proposed in section 7.3.4, iii) the grey scale and colour components of SIFT.

We evaluate the following fusion schemes: a) Bi-modal Second-order Occurrence Pooling ($r=2$) outlined in section 7.3.3 and referred to as *bi-modal* in the plots, b) the early fusion explained in section 7.3.1 and referred to as *early* in the plots, c) the late fusion explained in section 7.3.2 and referred to as *late*. Also, we often compare the classification performance on FK and VLAT, both employing the early fusion only. Moreover, for the proposed bi-modal fusion, equation (7.51) predicts 3 terms $\hat{h}_k^s$ that are weighted by $w^s=\binom{r}{s}^{\frac{1}{2}}(1-\beta)^{\frac{s}{2}}\beta^{\frac{r-s}{2}}$ in equation (7.51), where $s=0,...,2$. If $w^2 \ll w^0$ or $w^0 \ll w^2$, we reject all $\hat{h}_k^2$ or $\hat{h}_k^0$ (they become negligible) to shorten the signature.

Residual Descriptor is combined with SC, LLC, and LcSA by the bi-modal and late fusions on the PascalVOC07 set given $K=600$ dictionary atoms. Figure 7.10 (a) shows the baseline performance for Second-order Occurrence Pooling (*grey*). The late fusion (*late*) of the Residual Descriptor resulted in loss for SC and a marginal improvement for LLC and LcSA. This is expected as the residual codes are not associated in such a fusion neither with the corresponding descriptors nor the mid-level features (refer section 7.3.5 for the details). However, capturing co-occurrences of Residual Descriptors with the corresponding features (*bi-modal*) results in a significant gain of 0.8%, 1.6%, and 3.3% MAP for SC, LLC, and LcSA, respectively. The greater the quantisation loss for the

coder is, the greater the alleviating effect becomes. Note also that SC attains 66.2% MAP with the overall signature length $K^*=265356$. The same result was obtained in section 7.4.2 for the uni-modal second-order case given longer signature $K^*=320400$.

SPM and DoPM (the special case) proposed in section 7.3.4 were fused by Bi-modal Second-order Occurrence Pooling on the PascalVOC07 set. Figure 7.10 (b) demonstrates their performance (*bi-modally*) compared to SPM combined in an ordinary manner with Second-order Occurrence Pooling (*naively*). Bi-modally fused SPM scores 65.8% MAP giving a 0.8% improvement over the naively fused SPM which yields only 65.0% MAP. It also produces the signatures of length $K^*=510500$ (bi-modal case) compared to much longer 714780 (naive case). However, the uni-modal second-order case ($r=2$) from section 7.4.2 that employs SCC scores the highest. This suggests that SPM enhances the standard BoW ($r=1$) by extending its visual vocabulary (refer section 7.3.4 for the details). Once the visual vocabulary is extended by the co-occurrence statistics, the benefit of SPM becomes less obvious.



(*a*) VOC07, fusing colour (*b*) Flower102, fusing colour

Figure 7.11: Evaluation of Bi-modal Second-order Occurrence Pooling (*bi-modal*). The grey and opponent components of SIFT were fused in various ways given (*a*) PascalVOC07 and (*b*) Flower102 sets. The overall signature length K^* is indicated. Results for the early and late fusions from sections 7.3.1 and 7.3.2 are also provided for order $r=2$. Moreover, the early fusion was applied to FK, VLAT, and DoPM ($r=1$).

Fusing colour. The grey and opponent components of SIFT are fused now to obtain a further improvement of the classification results on three popular datasets. Figure 7.11 (a) introduces results attained by us on the PascalVOC07. The bi-modal fusion (*bi-modal*) scores 69.2% MAP for $K=800$ dictionary atoms. Note that one grey and one colour dictionary are used. This produces the signatures of length $K^*=960400$ as we rejected all $\hat{h}_k^2$ as explained earlier. The late fusion scores 68.6% MAP at its best for $K^*=640800$. This amounts to a 0.6% decline. The early fusion scores respectable 67.3% MAP for for $K=1000$ atoms that result in the signature length $K^*=500500$. Lastly, FK and VLAT yield 65.6% and 64.8% MAP, respectively.

Figure 7.11 (b) details results on the Flower102 set. The bi-modal fusion (*bi-modal*) scores 90.2% MAP for $K=800$ dictionary atoms and the signature length $K^*=960400$. The late fusion scores 89.3% MAP at its best for $K^*=640800$. This amounts to a 0.9% decline over the bi-modal approach. The early fusion scores respectable 89.4% MAP for for $K=1000$ atoms that result in the signature length $K^*=500500$. FK and

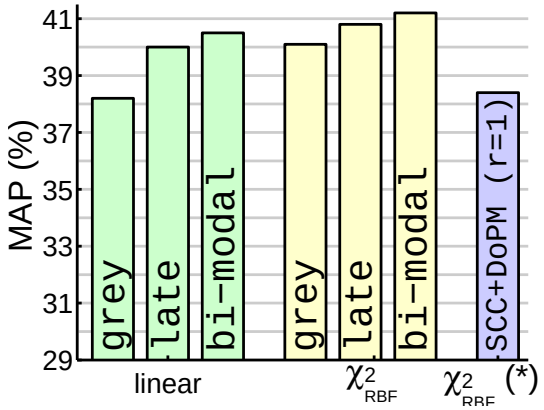


Figure 7.12: Evaluation of Uni-modal (*grey*) and Bi-modal (*bi-modal*) Second-order Occurrence Pooling (ImageCLEF11) given the linear and χ^2_{RBF} kernels. The late fusion of grey and colour SIFT components (*late*) is provided. A result on SCC and DoPM (*) for $r=1$ was taken from section 6.4.3.

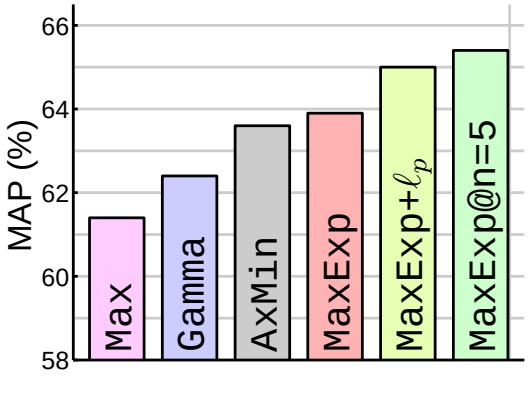


Figure 7.13: Evaluation of the pooling operators on the PascalVOC07 set. Max-pooling, Gamma, AxMin, MaxExp, MaxExp+ℓ_p, and @n were combined with Second-order Occurrence Pooling. SC with a dictionary of 600 atoms and the descriptor interdependency generalised operators from section 6.3.4 were used.

VLAT yield 89.3% and 88.7% MAP. The standard BoW ($r=1$) with DoPM yields 89.3% MAP for $K=4000$ atoms that result in the signature length $K^*=24000$. This represents a good trade-off between the classification scores and the signature lengths.

Figure 7.12 presents performance of Uni- and Bi-modal Second-order Occurrence Pooling on the ImageCLEF11 set. As ImageCLEF11 consists largely of abstract topics, *e.g.* *party life*, we also compare the classification performance of linear and χ^2_{RBF} kernels. The uni-modal, late, and the bi-modal approaches (*grey*, *late*, and *bi-modal* bars) score 38.2%, 40.0%, and 40.5% MAP given the linear kernel. $K=800$ atoms were used that produced the signature lengths $K^*=320400$, 640800, and 960400, respectively. A further improvement is observed for χ^2_{RBF} kernel with scores of 40.1%, 40.8%, and 41.2% MAP, respectively. This compares favourably to the late fusion of SCC and DoPM ($\chi^2_{RBF}(^*)$) given BoW ($r=1$) in section 6.4.3.

7.4.5 Evaluating the Pooling Operators

We conclude our evaluations with the classification results of Uni-modal Second-order Occurrence Pooling combined with a variety of pooling operators on the PascalVOC07 set. We use SC with $K=600$ dictionary atoms and SCC for spatial information.

Figure 7.13 shows that the best score of 65.4% MAP is obtained with the @n operator. Max-pooling is the weakest operator scoring 61.4% MAP. This amounts to a 4% gap in performance and is consistent with extensive comparisons of such operators provided in chapter 6. Moreover, this demonstrates importance of a pooling operator to the process of aggregation of the co-occurrences of visual words in the mid-level features.

7.5 Conclusions

This chapter proposes a theoretically derived framework that extends Bag-of-Words with the second- or higher-order statistics computed on the mid-level features. We term these approaches as Second- and Higher-order Occurrence Pooling. According to our evaluations, Uni-modal Second-order Occurrence Pooling offers the best trade-off between the tractability of coding, the length of signatures, and the classification quality

grey SIFT	VOC07	Caltech101 (15 img.)	Caltech101 (30 img.)	CLEF11
Uni-modal ($r=2$)	**66.2**	**76.6 ± 0.5**	**83.6 ± 0.4**	**40.1**
FV	64.3	75.7 ± 0.5	82.2 ± 0.4	-
VLAT	63.7	74.2 ± 0.4	81.1 ± 0.7	-
SCC ($r=1$)	62.4	72.0 ± 0.3	77.7 ± 0.7	-
SPM ($r=1$)	62.8	74.9 ± 0.4	81.5 ± 0.5	-
DoPM ($r=1$)	63.6	-	-	-

grey+colour	VOC07	Flower102	CLEF11
Bi-modal ($r=2$)	**69.2**	**90.2**	**41.2**
Early ($r=2$)	67.3	89.4	-
Late ($r=2$)	68.6	89.3	40.8
FV	65.6	89.3	-
VLAT	64.8	88.7	-
DoPM ($r=1$)	-	89.3	-

Table 7.2: Summary of the best results from this chapter. The signature lengths for the results in this table vary. See figures 7.8-7.12 for a fair and exact comparison.

method	VOC07	method	Flower102
[Sánchez et al., 2012]	66.3	[Awais et al., 2011b]	80.3
[Gong et al., 2009]	64.0	[Awais et al., 2011a]	75.7
[Zhou et al., 2010]	64.0	[Yuan and Yan, 2010]	74.1
[Perronnin et al., 2010]	60.3	[Zhang et al., 2012]	76.9

method	Caltech101 (30 img.)	method	CLEF11
[Duchenne et al., 2011]	80.3 ± 1.2	[Binder et al., 2011]	38.8
[Bosch et al., 2007]	81.3 ± 0.8	[Su and Jurie, 2011]	38.2
[Kulkarni and Li, 2011]	83.3	[Avila et al., 2012]	38.4
[Yang et al., 2012a]	84.3	Chapter 6	38.4

Table 7.3: Summary of the best results from other studies.

for the grey scale descriptors. Such an approach is demonstrated to outperform the standard BoW with various Pyramid Matching schemes, Fisher Vector Encoding, and closely related Vector of Locally Aggregated Tensors. Evaluations were performed in a common testbed on the PascalVOC07, Caltech101, and ImageCLEF11 sets. Moreover, care was taken to compare the prior work regarding the coding and pooling techniques and determine their suitability for the proposed framework. Sparse Coding and the @n pooling operator are found to be the best performers.

To benefit from the multi-modal nature of visual concepts, a bi-modal extension is formulated. We term such an approach as Bi-modal Second-order Occurrence Pooling. Its extensions to the multi-modal and higher-order variants are suggested. The proposed bi-modal approach predicts existence of cross-modal statistics. Their importance is demonstrated with extended Pyramid Matching schemes and Residual Descriptor exploiting the quantisation effects in coding.

Such a bi-modal variant is also compared extensively to the outlined early and late fusions performed between the grey and colour components of descriptors on the standard BoW, Second-order Occurrence Pooling, Fisher Vector Encoding, and Vector of Locally Aggregated Tensors. For this purpose, the PascalVOC07, Flower102, and ImageCLEF11 set are used. Given a common testbed, the proposed Bi-modal Second-order Occurrence Pooling is shown to outperform other approaches. Table 7.2 lists the best results from our study. See appendix A.4 for a statistical significance test. For comparison, we provide a selection of the best results from other studies in table 7.3.

Possible extensions of this work include compression of the image signatures to limit their length. FK from [Jégou et al., 2012] and VLAT from [Negrel et al., 2012] already exploit such a compression.

Chapter 8

Conclusions

In this thesis, we have studied various steps that constitute on the Bag-of-Words model. This resulted in a number of image representations with an increased invariance to the repeatable visual stimuli, also known as *burstiness* of features. As a result of these investigations, the following new methods have been contributed:

- A segmentation-based interest point detector to extract salient keypoints from informative regions of images.
- A segmentation-based semi-local image descriptor to encode semi-local image structures and ignore uniform appearances.
- An optimisation scheme for the Soft Assignment coding step to minimise its quantisation loss.
- An alternative approach to SPM that introduces the spatial information to the classification process at the descriptor level, called Spatial Coordinate Coding.
- Two alternative Pyramid Matching schemes that exploit dominant edge and colour bias in images, called Dominant Angle and Colour Pyramid Matching, respectively.
- New mid-level feature pooling approaches that take into account the descriptor interdependence and other phenomena, *e.g. the leakage* resulting from the coding step in Bag-of-Words.

- An aggregation step over co-occurrences of visual words in mid-level features called Higher-order Occurrence Pooling. This can be seen as a simple approach which increases the numbers of visual words in a given dictionary.

The following list is a summary of the contributions and conclusions from each chapter:

- In chapter 2, various unsupervised segmentations were evaluated with aim to extract salient repeatable keypoints from them. The most convex and concave points along segment boundaries were located with the proposed SUSAN algorithm. They proved to be stable and repeatable under various photometric and geometric variations. Moreover, they also resulted in better classification scores compared to the dense sampling strategy. As impact of keypoints from large segments was diminished, this suggested that aggregating multiple contributions from uniform areas of images into the final representations must be detrimental to visual categorisation. This was confirmed, as adding back the dense sampled points from such areas decreased back the results. The proposed detector instead delivered keypoints from areas where the biggest changes in appearance took place, as dictated by the segmentation maps. Therefore, we concluded that the local image descriptors extracted at these locations were more distinctive compared to descriptors resulting from dense sampling. However, a minor drawback of working with the unsupervised segmentation algorithms is that they fail to enclose textured regions into large segments. Hence, they cannot be easily used to diminish the impact of repeatable texture patterns.

- In chapter 3, segmentation-based semi-local image descriptors were designed and studied. They resulted in compact, relatively low dimensional image representations, that are especially highly suitable for large scale experiments. Segmentation maps were investigated for their ability of delivering the robust spatial hypotheses of object parts. The local image descriptors like SIFT employ rigid spatial bins. This means that SIFT tends to somewhat blend foregrounds and backgrounds, and that spatial bins are never fully aligned to the complex shapes of objects. Our approach resulted in semi-local image representations built from pairs of adjacent segments. This captured only neighbouring object parts that are more

likely to repeat than complex representations across images of the same category. On the other hand, they may be less discriminative. Therefore, various image statistics were extracted from image regions indicated by segments. Moreover, as such segments cover entire images, all image regions were represented well unlike in case of typical interest point detectors that occasionally contribute very few keypoints. We also note that the large uniform areas in images, as dictated by the segmentation maps, contributed fewer vectors compared to dense sampling. Therefore, this diminished impact of uninformative appearances and resulted in semi-local compact representations that outperformed SIFT.

- In chapter 4, an intuitive coding approach called Soft Assignment was investigated in the context of Linear Coordinate Coding methods that are popular due to their robustness in visual categorisation. The SA coder embeds the local image descriptors into a given vocabulary space in order to represent images by the compact vectors. This process is however impaired by the quantisation effects that take place during the coding procedure. We presented a novel method for finding the so-called smoothing factor of the SA model by minimising the quantisation loss, typically employed by the LCC family. We observed that minimising the quantisation loss for SA correlated strongly with peaks in the classification scores. We conclude that the smoothing parameter selected in such a manner helps linearise the SA model. Moreover, we note that a large quantisation loss in the coding step has a detrimental impact on the classification process.

- In chapter 5, an alternative approach to Spatial Pyramid Matching was proposed. A trade-off between visual appearance and spatial bias, called Spatial Coordinate Coding, was implemented on the coding level. This was achieved by minimising two terms for the quantisation loss in the SC coder. Similar considerations applied to SA and resulted in an observation that the SCC scheme can be simply implemented at the descriptor level by concatenating descriptors with the corresponding spatial spatial locations. Moreover, the proposed method outperformed SPM and resulted in significantly smaller image representations. This enabled investigations into Pyramid Matching applied to cues other than spatial informa-

tion. Dominant edges in images were proposed as a good source of bias in images. Therefore, the Dominant Angle cues implemented in SIFT were employed in order to form DoPM. Quantising DA at multiple levels of coarseness resulted in improved classification performance over simply using DA at the descriptor level. Next, similar ideas were successfully applied to the colour. Based on experimental results, we conclude that the spatial bias is best exploited at the descriptor level while dominant edges benefit more form Pyramid Matching. The proposed SCC and DoPM were additionally used and compared in various classification scenarios in the remaining chapters of this thesis.

- Chapter 6 introduced the major contributions that address the phenomenon of repeatable visual patterns. Evaluations from chapters 2 and 3 strongly suggested that reducing contributions from large uniform regions in images can increase the classification performance. However, segmentations used for that purpose suffered from inability to cope with textures and the structural noise. Therefore, dense sampling was employed, and the pooling step that aggregates the mid-level features into the image signatures was investigated. The aggregation step builds statistics about occurrences of visual words in each image. Therefore, a family of likelihood inspired operators were generalised by us to account for the descriptor interdependency. This resulted in a robust estimator of probability of *at least one particular visual word being present in an image*. Such a pooling step acted as a reliable detector of visual prototypes. Moreover, instead of counting the visual appearances of any given type, and therefore quantifying areas covered by them, this operator just registered how likely it was for the prototype to be contained by the image. This significantly improved the classification results. Moreover, a pooling extension called the $@n$ operator was proposed to further cope with a coding noise called *the leakage*. Other contributions include a fast coding technique called Approximate Locality-constrained Soft Assignment, its optimisation step that minimises its quantisation loss, and a speed-wise improvement of coding based on Spill Trees. Furthrmore, interaction between the coding and pooling steps was demonstrated in numerous practical evaluations never undertaken by others on such a scale. It revealed the best coding and pooling operators for visual

categorisation, and demonstrated that SCC and DoPM also benefit from these operators. The state-of-the-art results were attained. We conclude that both coding and pooling steps have a major impact on visual categorisation. Also, the pooling operator should account for the phenomena taking place in a coder. To conclude, the pooling step has an immense ability to diminish the impact of statistically unpredictable repetitions of visual patterns.

- Chapter 7 is the culmination of the investigations of the BoW models. Typically, a pooling operator aggregates occurrences of visual words represented by coefficients of each mid-level feature vector associated with the descriptors. However, approaches such as Fisher Vector Encoding have outperformed BoW based on SC, LLC, and similar coders. This chapter analysed various discrepancies between typical BoW and FK. It was concluded that FK differs in its coding step, employs the second-order statistics for the image representations, and exploits a likelihood inspired pooling step. Therefore, the differences between both models were addressed. The main contribution of this chapter lies in equipping the BoW model with the second- or higher-order statistics. Specifically, we proposed the aggregation step over co-occurrences of visual words in mid-level features called Second-order Occurrence Pooling. Second- and Higher-order Occurrence Pooling were analytically derived based on linearisation of so-called Minor Polynomial Kernel. Generalisation to various pooling operators was explored: Max-pooling, Analytical pooling, and a highly effective trade-off between Max-pooling and Analytical pooling called the @n operator from chapter 6. Such an equipped BoW attained significant improvements over Fisher Vector Encoding. Having analysed the nature of co-occurrences, we concluded that they simply increase the resolution of a given visual vocabulary. Furthermore, as the classification process often benefits from fusing multiple complementary modalities, *e.g.* the grey scale and colour descriptors, we developed a bi- and multi-modal coding for two or more coders. This represents an extension of Second- and Higher-order Occurrence Pooling. It is demonstrated extensively by combining both the grey scale and colour mid-level features that such an approach outperforms naive fusing schemes. Moreover, the SPM scheme for BoW and other similar methods are

shown as special cases of this approach. The second-order statistics collected by SPM explain why it has been such a remarkable performer. Lastly, a Residual Descriptor that exploits the quantisation loss of the coding step was designed to work with the bi-modal extension. It thrived on the quantisation loss of coders. To conclude, various comparisons to the state-of-the-art systems show that the proposed model outperformed them significantly on various datasets.

To conclude, Bag-of-Words is a robust and flexible model for visual categorisation. Its various components can be adapted to specific tasks, *e.g.* Visual Object Category Recognition or Visual Concept Detection. We have observed that bias in images such as dominant orientations of edges, dominant colours, or spatial locations can be beneficial in recognition. Therefore, descriptors which are only partially invariant to orientations of edges produced very good results. However, it remains an open question, whether representations designed to exploit bias in images can generalise sufficiently well between different datasets. Moreover, we have observed that the variance introduced by the repeatable visual patterns can be suppressed at various stages of the BoW model. The most effective strategy proposed in this thesis is pooling designed to cope with the descriptor interdependence. Nonetheless, experiments with segmentation-based interest points and semi-local descriptors have also shown a promise. More importantly, as the variance from repeatable visual stimuli is limited, the classification results improve. This suggest that the local image descriptors produce very distinct representations of objects (or their parts), *e.g.* more complex ensembles of descriptors may be redundant.

Furthermore, we have observed that minimising the quantisation error during the coding step facilitates better results. This can be explained by preventing a loss of information in the coding step. Moreover, we have demonstrated that the pooling step is prone to a loss of information. Often, the best quantisation may result in the features that are not distinct enough for the pooling step to produce meaningful image signatures. Therefore, we have introduced the second-order statistics to represent the content from the coding step robustly. This reduces uncertainty introduced by the pooling step. Therefore, the obtained image signatures become more distinctive. We note that the BoW model requires more studies due to complex interactions between its components.

8.1 Further Directions

Bag-of-Words includes several components that often constitute separate directions of research. Robust local image descriptors exploiting curvature of objects, as well as accounting for various transformations, could improve the classification results. For instance, we doubled the number of training images by left-right flipping operation. This provided invariance to flipping at a cost of additional computations that could be avoided. Moreover, as the proposed segmentation-based descriptors proved to be particularly suited in large scale experiments due to their relatively low dimensionality, we expect that such a representation may be particularly beneficial given the spatio-temporal data for the action and event recognition. Moreover, note that SIFT descriptors apply by default a predefined threshold to strong gradients to decrease their impact. This procedure resembles the proposed AxMin correction. Therefore, the notion of *at least one particular visual word being present in an image* can be applied to the descriptors to prevent *burstiness* of image gradient: a statistical uncertainty of evidence of an edge.

The coding approaches are being constantly improved. According to our evaluations, SC is the best performing coder. However, its speed is prohibitive given a large visual dictionary. Criteria optimised by SC could be relaxed and a fast approximation devised. Another promising direction in coding is a supervised strategy for learning the manifold structure. Current approaches are only approximately respecting the underlying manifold by the globally imposed notion of locality. Moreover, invariance to the scale, rotation, and affine transformations of visual appearances can be learnt from the annotated data to link visual prototypes in a dictionary which currently may be replicated many times because local image descriptors are not fully invariant to various photometric and geometric transformations. Another direction of research on coding and dictionary learning could address how to robustly generalise a learned model between various datasets. This could be achieved by: i) studying a manifold resulting from a design of the descriptor representation, ii) studying difference in image bias between datasets, iii) employing transfer learning to small datasets to better approximate a true distribution.

The pooling step has been only recently emphasised as an important part of the BoW model. Currently, a couple of parameters of pooling require cross-validation. An interesting route would be to learn these parameters, perhaps even one per visual word, by fusing pooling with the classifier. As different visual words exhibit different levels of *burstiness* in collections of images, learning these parameters is a promising direction. An ensemble learning to select appropriate pooling method per visual word is also possible. Furthermore, as some visual words are strongly correlated, *e.g.* the sky and the sun, approaches to decorrelation of visual prototypes could further enhance performance. Whitening PCA applied to the image signatures can reduce redundancy. More importantly, the variance introduced by the repeatable visual patterns can be suppressed at various stages in the BoW model. Wider studies are required to understand how to best decrease it while reducing classification complexity and maximising the classification performance. Another line of investigations concerns fast semi-supervised interest point detectors and segmentations that can learn characteristics of the most discriminative and repeatable regions for visual categorisation.

Applying the second- or higher-order statistics has been demonstrated as a way of extending the visual dictionary. Alternative approaches to partitioning the descriptor space are of great interest, for instance combining FK with radial zones defined over Gaussian components. Moreover, as the second- and higher-order statistics have emerged to perform multi-modal fusion by schemes such as SPM, another great possibility is to investigate how these statistics can enrich classification in spatio-temporal and audio-visual domains, possibly further enhanced by the textual representations.

In the long term, attribute learning based on the proposed BoW model could result in a greater sensitivity to various objects and provide improved representations for complex visual concepts. For instance, projections on the second-order image signatures produced from the linked visual dictionary could be learnt to reflect various attributes. Then a scoring function based on such a representation can be designed to more accurately describe the content of images from various sources. Therefore, it can be employed to perform an accurate transfer learning by mining the internet resources.

Appendix A

A.1 Analytical Similarity of LcSA and LLC

Now, the analytical similarity between LcSA and LLC will be shown. The solution to Approximate Locality-constrained Linear Coding from equation (6.8), but without the non-negativity constraint, is given in [Wang et al., 2010]. It can be expressed as:

$$\begin{aligned} \boldsymbol{C} &= \left(\mathcal{M}^{'} - \mathbf{1}^{T}\boldsymbol{x}\right)^{T} \cdot \left(\mathcal{M}^{'} - \mathbf{1}^{T}\boldsymbol{x}\right) \\ \bar{\boldsymbol{\phi}} &= (\boldsymbol{C} + \lambda \cdot \boldsymbol{I})^{-1} \cdot \mathbf{1} \\ \boldsymbol{\phi} &= \bar{\boldsymbol{\phi}} / \mathbf{1}^{T}\bar{\boldsymbol{\phi}} \end{aligned} \tag{A.1}$$

Symbol $\boldsymbol{I}$ denotes the identity matrix, λ is a small regularisation constraint, *e.g.* $\lambda = 10^{-5}$, $\mathbf{1}$ is a vector with all coefficients equal 1, and symbol $\boldsymbol{x}$ is a descriptor to code. Moreover, $\boldsymbol{C}$ is a covariance matrix, $\mathcal{M}^{'}$ is a matrix storing a localised visual vocabulary such that anchors $\boldsymbol{m}_1, ..., \boldsymbol{m}_l$ from dictionary $\mathcal{M}$ form its columns. These anchors are the l-nearest anchors of descriptor $\boldsymbol{x}$ found with the NN search. Finally, $\boldsymbol{\phi}$ is a resulting mid-level code.

By assuming that matrix $\boldsymbol{C}$ has all off-diagonal elements equal 0, we turn inversion of $(\boldsymbol{C} + \lambda \cdot \boldsymbol{I})$ into a simple element-wise division:

$$(\boldsymbol{C} + \lambda \cdot \boldsymbol{I})^{-1} = \begin{bmatrix} \frac{1}{(\boldsymbol{m}_1 - \boldsymbol{x})^2 + \lambda} & 0 & 0 \\ 0 & ... & 0 \\ 0 & 0 & \frac{1}{(\boldsymbol{m}_l - \boldsymbol{x})^2 + \lambda} \end{bmatrix} \tag{A.2}$$

We note that $\left(\boldsymbol{m}_{l'} - \boldsymbol{x}\right)^2 = \left(\boldsymbol{m}_{l'} - \boldsymbol{x}\right)^{T} \cdot \left(\boldsymbol{m}_{l'} - \boldsymbol{x}\right)$. Furthermore, $\bar{\boldsymbol{\phi}}$ becomes simplified to the following expression:

$$\bar{\boldsymbol{\phi}} = \left[\frac{1}{(\boldsymbol{m}_1 - \boldsymbol{x})^2 + \lambda}, ..., \frac{1}{(\boldsymbol{m}_l - \boldsymbol{x})^2 + \lambda}\right]^{T} \tag{A.3}$$

Moreover, applying the final step such as $\boldsymbol{\phi}=\bar{\boldsymbol{\phi}}/\mathbf{1}^T\bar{\boldsymbol{\phi}}$ results in expression:

$$\phi_{l'} = \frac{\dfrac{1}{\left(\boldsymbol{m}_{l'}-\boldsymbol{x}\right)^2+\lambda}}{\displaystyle\sum_{\boldsymbol{m}'\in\mathcal{M}'}\frac{1}{\left(\boldsymbol{m}'-\boldsymbol{x}\right)^2+\lambda}} \tag{A.4}$$

The responses of model are computed over $l^{'}=1,...,l$. Such a solution is very similar analytically to the LcSA model from equation (6.9) that is expressed by the ratio of Gaussian functions and therefore further simplified to:

$$\phi_{l'} = \frac{\dfrac{1}{\exp\left(\left(\boldsymbol{m}_{l'}-\boldsymbol{x}\right)^2/\left(2\sigma^2\right)\right)}}{\displaystyle\sum_{\boldsymbol{m}'\in\mathcal{M}'}\frac{1}{\exp\left(\left(\boldsymbol{m}'-\boldsymbol{x}\right)^2/\left(2\sigma^2\right)\right)}} \tag{A.5}$$

Equations (A.4) and (A.5) can be shown to result in approximately similar solutions if λ and σ are chosen appropriately. We verified this with the second-order Taylor expansion. For instance, we assumed two $1D$ anchors such that $m_1=1$ and $m_2=-1$, $\lambda=10^{-5}$ and $\sigma=0.25$. Equations (A.4) and (A.5) were expanded around point $\bar{x}=0$ which resulted in $\frac{10^5}{10^5+1}x+\frac{1}{2}$ and $x+\frac{1}{2}$, respectively.

A.2 Optimisation of LcSA cost

Optimising the cost posed in equation (6.10) in order to find parameters $(\sigma,\ l)$ can be performed by a coordinate-descent solver. This requires computing both first and second derivatives with respect to the parameters. The gradient is approximated by:

$$\frac{\partial\xi^2}{\partial\sigma} \approx \frac{\xi^2(\sigma+\Delta\sigma,l)-\xi^2(\sigma-\Delta\sigma,l)}{2\Delta\sigma} \tag{A.6}$$

$$\frac{\partial\xi^2}{\partial l} \approx \frac{\xi^2(\sigma,l+\Delta l)-\xi^2(\sigma,l-\Delta l)}{2\Delta l} \tag{A.7}$$

Parameter $\Delta\sigma$ depends on the descriptors used in the experiments outlined in the next section. It determines the quality of approximation of the gradient and is set arbitrarily to 1 and 0.001 for descriptors such that $\|\boldsymbol{x}\|_2=255$ and $\|\boldsymbol{x}\|_2=1$, respectively. Parameter Δl is set to 1 because l-nearest anchors is a positive integer value. Hessian

matrix increases the speed of convergence for coordinate-descent solvers. It is given by:

$$\frac{\partial^2 \xi^2}{\partial \sigma^2} \approx \frac{\xi^2(\sigma + \Delta\sigma, l) + \xi^2(\sigma - \Delta\sigma, l) - 2\xi^2(\sigma, l)}{(\Delta\sigma)^2} \tag{A.8}$$

$$\frac{\partial^2 \xi^2}{\partial l^2} \approx \frac{\xi^2(\sigma, l + \Delta l) + \xi^2(\sigma, l - \Delta l) - 2\xi^2(\sigma, l)}{(\Delta l)^2} \tag{A.9}$$

$$\frac{\partial^2 \xi^2}{\partial \sigma \partial l} \approx \frac{\xi^2(\sigma+\Delta\sigma, l+\Delta l) + \xi^2(\sigma-\Delta\sigma, l-\Delta l) - \xi^2(\sigma+\Delta\sigma, l-\Delta l) - \xi^2(\sigma-\Delta\sigma, l+\Delta l)}{4\Delta\sigma\Delta l} \tag{A.10}$$

The first step in this algorithm is an efficient search for the l-nearest anchors from dictionary $\mathcal{M}$ for each descriptor that is selected for the optimisation procedure and contained in the descriptor set $\mathcal{X}$. Note that both anchors and descriptors are column vectors in matrices $\mathcal{M} \in \mathbb{R}^{D \times K}$ and $\mathcal{X} \in \mathbb{R}^{D \times N}$, respectively. Symbols D, K, and N are the descriptor dimensionality, the number of atoms in the dictionary, and the number of descriptors selected for optimisation. First, the squared ℓ_2 norm is decomposed to obtain matrix $\mathcal{D} \in \mathbb{R}^{N \times D}$ containing the squared distances between descriptors and anchors:

$$\mathcal{D} = \|\mathcal{X}\|^2_{2CW} \cdot \mathbf{1}^T - 2\mathcal{X}^T\mathcal{M} + \mathbf{1} \cdot \|\mathcal{M}\|^2_{2CW}{}^T \tag{A.11}$$

Operators $\|\mathcal{X}\|^2_{2CW} \in \mathbb{R}^N$ and $\|\mathcal{M}\|^2_{2CW} \in \mathbb{R}^K$ compute the squared ℓ_2 norm per column vector, as indicated by CW, and result in a vector of norms each, respectively. Vector $\mathbf{1}$ consists of coefficients equal 1. It is used to replicate vectors $\|\mathcal{X}\|^2_{2CW}$ and $\|\mathcal{M}\|^2_{2CW}$ along rows and columns, respectively. This is accomplished by applying the outer product. Matrix $\mathcal{D}$ is sorted by the partial sort algorithm along rows to find the l-nearest anchors for each descriptor.

There exist 9 different terms of function ξ^2 for all combinations of its input parameters: $\{\sigma-\Delta\sigma,\ \sigma,\ \sigma+\Delta\sigma\} \times \{l-\Delta l,\ l,\ l+\Delta l\}$. These 9 terms have to be computed in order to estimate the approximate first and second derivatives given by equations (A.6-A.10). Evaluating ξ^2 in a naive manner entire 9 times is computationally costly. A fast algorithm that exploits redundancy in computing the membership probabilities from equation (6.9) used by the cost in equation (6.10) is presented on the following page, referred to as algorithm 1. The provided snippet takes current σ and l from the solver and returns matrix $\boldsymbol{\xi}^2 \in \mathbb{R}^{3 \times 3}$ that is required to compute the first and second

derivatives in equations (A.6-A.10):

Data: $\mathcal{X}$, $\mathcal{M}$, σ, l, $\Delta\sigma$, Δl

Result: $\boldsymbol{\xi}^2 \in \mathbb{R}^{3\times 3}$ such that

$$\boldsymbol{\xi}^2 = \begin{bmatrix} \xi^2(\sigma-\Delta\sigma, l-\Delta l) & \xi^2(\sigma, l-\Delta l) & \xi^2(\sigma+\Delta\sigma, l-\Delta l) \\ \xi^2(\sigma-\Delta\sigma, l) & \xi^2(\sigma, l) & \xi^2(\sigma+\Delta\sigma, l) \\ \xi^2(\sigma-\Delta\sigma, l+\Delta l) & \xi^2(\sigma, l+\Delta l) & \xi^2(\sigma+\Delta\sigma, l+\Delta l) \end{bmatrix}$$

initialisation:

$\boldsymbol{\xi}^2 = \mathbf{0}^{3\times 3}$ (3×3 matrix filled with zeros)

$\mathcal{D} = \|\mathcal{X}\|^2_{2CW} \cdot \mathbf{1}^T - 2\mathcal{X}^T\mathcal{M} + \mathbf{1} \cdot \|\mathcal{M}\|^2_{2CW}{}^T$ (compute $\mathcal{D}$ as explained)

foreach $\boldsymbol{x} \in \mathcal{X}$ **do**

- extract the $l+\Delta l$-nearest distances from $\mathcal{D}$ into vector $\boldsymbol{d} \in \mathbb{R}^{l+\Delta l}$
- extract the $l+\Delta l$-nearest anchors from $\mathcal{M}$ into matrix $\boldsymbol{M} \in \mathbb{R}^{D\times(l+\Delta l)}$
- also $\boldsymbol{M}' = \&\boldsymbol{M}(:, 1\!:\!end-1) \in \mathbb{R}^{D\times l}$, $\boldsymbol{M}'' = \&\boldsymbol{M}'(:, 1\!:\!end-1) \in \mathbb{R}^{D\times(l-\Delta l)}$

  (& denotes referencing rather than copying)

$\boldsymbol{d} := -\boldsymbol{d}$

- form $\bar{\boldsymbol{d}}_1$, $\bar{\boldsymbol{d}}_2$, $\bar{\boldsymbol{d}}_3$ such that $\bar{\boldsymbol{d}}_1 = \boldsymbol{d}/(\sigma-\Delta\sigma)^2$, $\bar{\boldsymbol{d}}_2 = \boldsymbol{d}/\sigma^2$, $\bar{\boldsymbol{d}}_3 = \boldsymbol{d}/(\sigma+\Delta\sigma)^2$
- form $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, $\boldsymbol{e}_3$ such that $\boldsymbol{e}_1 = \exp(\bar{\boldsymbol{d}}_1)$, $\boldsymbol{e}_2 = \exp(\bar{\boldsymbol{d}}_2)$, $\boldsymbol{e}_3 = \exp(\bar{\boldsymbol{d}}_3)$
- according to equation (6.9), we compute 9 sums s for the membership probability denominator, the remaining 6 enumerator vectors $\boldsymbol{e}$, and 9 ratios of Gaussians $\boldsymbol{r}$, then 9 residual approximations $\hat{\boldsymbol{x}}$, and 9 costs $\bar{\xi}^2$:

for $i = 1$ *to* 3 **do**

$s_i = \sum_{j=1}^{l+\Delta l} e_{ji}$, $s_i' = s_i - \sum_{j=l+1}^{l+\Delta l} e_{ji}$, $s_i'' = s_i' - \sum_{j=l-\Delta l}^{l} e_{ji}$ (efficient sums)

$\boldsymbol{e}_i' = \&\boldsymbol{e}_i(1\!:\!end-1)$, $\boldsymbol{e}_i'' = \&\boldsymbol{e}_i'(1\!:\!end-1)$ (enumerator vectors)

$\boldsymbol{r}_i = \frac{\boldsymbol{e}_i}{s_i}$, $\boldsymbol{r}_i' = \frac{\boldsymbol{e}_i'}{s_i'}$, $\boldsymbol{r}_i'' = \frac{\boldsymbol{e}_i''}{s_i''}$ (ratios of Gaussians)

$\hat{\boldsymbol{x}}_i = \boldsymbol{M} \cdot \boldsymbol{r}_i$, $\hat{\boldsymbol{x}}_i' = \boldsymbol{M}' \cdot \boldsymbol{r}_i'$, $\hat{\boldsymbol{x}}_i'' = \boldsymbol{M}'' \cdot \boldsymbol{r}_i''$ (linear approximations)

$\bar{\xi}^2_{3i} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}_i\|_2^2$, $\bar{\xi}^2_{2i} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}_i'\|_2^2$, $\bar{\xi}^2_{1i} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}_i''\|_2^2$ (costs per $\boldsymbol{x}$)

end

$\boldsymbol{\xi}^2 := \boldsymbol{\xi}^2 + \bar{\boldsymbol{\xi}}^2$ (update the final costs)

end

Algorithm 1: Fast computations of 9 cost coefficients for the partial derivatives.

A.3 Lower Bound of BoW for @n Operator

The standard BoW with the Avg@n operator and Polynomial Kernel of degree r is given in equation (A.12) which is then expanded in equation (A.13) and simplified to a dot product between two vectors in equation (A.14). Such an expression forms a linear kernel. A simple lower bound of equation (A.13) is proposed in equation (A.15). Note that it represents Higher-order Occurrence Pooling with the Avg@n operator further linearised to a dot product between two vectors in equation (A.16).

$$Ker\left(\boldsymbol{\Phi}, \bar{\boldsymbol{\Phi}}\right) = \left\langle \hat{\boldsymbol{h}}, \bar{\hat{\boldsymbol{h}}} \right\rangle^r, \text{ and } \begin{cases} \hat{h}_k = \operatorname{avg\,srt}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}, @n\right) \\ \bar{\hat{h}}_k = \operatorname{avg\,srt}\left(\{\bar{\phi}_{kn}\}_{\bar{n}\in\bar{\mathcal{N}}}, @n\right) \end{cases}$$

$$= \left(\sum_{k=1}^{K} \operatorname{avg\,srt}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}\right) \cdot \operatorname{avg\,srt}\left(\{\bar{\phi}_{k\bar{n}}\}_{\bar{n}\in\bar{\mathcal{N}}}\right) \right)^r \tag{A.12}$$

$$= \sum_{k^{(1)}=1}^{K} \dots \sum_{k^{(r)}=1}^{K} \Bigg(\operatorname{avg\,srt}\left(\{\phi_{k^{(1)}n}\}_{n\in\mathcal{N}}\right) \cdot \dots \cdot \operatorname{avg\,srt}\left(\{\phi_{k^{(r)}n}\}_{n\in\mathcal{N}}\right) \cdot \tag{A.13}$$

$$\cdot \operatorname{avg\,srt}\left(\{\bar{\phi}_{k^{(1)}\bar{n}}\}_{\bar{n}\in\bar{\mathcal{N}}}\right) \cdot \dots \cdot \operatorname{avg\,srt}\left(\{\bar{\phi}_{k^{(r)}\bar{n}}\}_{\bar{n}\in\bar{\mathcal{N}}}\right) \Bigg)$$

$$= \left\langle u_{:}^{*}\left[\otimes_r \underset{n\in\mathcal{N}}{\operatorname{avg\,srt}}\left(\boldsymbol{\phi}_n, @n\right) \right], u_{:}^{*}\left[\otimes_r \underset{\bar{n}\in\bar{\mathcal{N}}}{\operatorname{avg\,srt}}\left(\bar{\boldsymbol{\phi}}_{\bar{n}}, @n\right) \right] \right\rangle \tag{A.14}$$

$$\geq \frac{1}{@n^{2r-2}} \sum_{k^{(1)}=1}^{K} \dots \sum_{k^{(r)}=1}^{K} \Bigg(\operatorname{avg\,srt}(\{\phi_{k^{(1)}n} \cdot \dots \cdot \phi_{k^{(r)}n}\}_{n\in\mathcal{N}}, @n) \cdot \tag{A.15}$$

$$\cdot \operatorname{avg\,srt}\left(\{\phi_{k^{(1)}\bar{n}} \cdot \dots \cdot \bar{\phi}_{k^{(r)}\bar{n}}\}_{\bar{n}\in\bar{\mathcal{N}}}, @n\right) \Bigg)$$

$$= \frac{1}{@n^{2r-2}} \left\langle \underset{n\in\mathcal{N}}{\operatorname{avg\,srt}}\left[u_{:}^{*}(\otimes_r \boldsymbol{\phi}_n), @n\right], \underset{\bar{n}\in\bar{\mathcal{N}}}{\operatorname{avg\,srt}}\left[u_{:}^{*}\left(\otimes_r \bar{\boldsymbol{\phi}}_{\bar{n}}\right), @n\right] \right\rangle \tag{A.16}$$

Indexes $\mathcal{N}$ and $\bar{\mathcal{N}}$ indicate the mid-level features $\boldsymbol{\phi}_n$ and $\bar{\boldsymbol{\phi}}_{\bar{n}}$ from any two chosen images. Notation $\operatorname{avg\,srt}\left(\{\phi_{kn}\}_{n\in\mathcal{N}}, @n\right)$ denotes averaging over the top $@n$ coefficients from set $\{\phi_{kn}\}_{n\in\mathcal{N}}$. Moreover, $\underset{n\in\mathcal{N}}{\operatorname{avg\,srt}}\left(\boldsymbol{\phi}_n, @n\right)$ denotes averaging over the top $@n$ coefficients from set $\{\phi_{1n}\}_{n\in\mathcal{N}}$, then set $\{\phi_{2n}\}_{n\in\mathcal{N}}$, and so on. This results in the following vector:

$$\underset{n\in\mathcal{N}}{\operatorname{avg\,srt}}\left(\boldsymbol{\phi}_n, @n\right) = \left[\operatorname{avg\,srt}\left(\{\phi_{1n}\}_{n\in\mathcal{N}}, @n\right), \operatorname{avg\,srt}\left(\{\phi_{2n}\}_{n\in\mathcal{N}}, @n\right), \dots\right]^T$$

Equation (A.13) is an upper bound of equation (A.15) as the following inequality holds:

$$\begin{aligned}
&\operatorname{avg}\operatorname{srt}\left(\left\{\phi_{k^{(1)}n}\right\}_{n\in\mathcal{N}}\right)\cdot\ldots\cdot\operatorname{avg}\operatorname{srt}\left(\left\{\phi_{k^{(r)}n}\right\}_{n\in\mathcal{N}}\right) && \text{(A.17)}\\
&=\frac{1}{@n}\sum_{n\in\mathcal{N}^*_{k^{(1)}}}\phi_{k^{(1)}n}\cdot\ldots\cdot\frac{1}{@n}\sum_{n\in\mathcal{N}^*_{k^{(r)}}}\phi_{k^{(r)}n}\\
&=\frac{1}{@n^r}\sum_{n\in\mathcal{N}^*_{k^{(1)}}}\phi_{k^{(1)}n}\cdot\ldots\cdot\sum_{n\in\mathcal{N}^*_{k^{(r)}}}\phi_{k^{(r)}n}\\
&\geq\frac{1}{@n^{r-1}}\operatorname{avg}\operatorname{srt}(\{\phi_{k^{(1)}n}\cdot\ldots\cdot\phi_{k^{(r)}n}\}_{n\in\mathcal{N}},@n)\\
&=\frac{1}{@n^r}\sum_{n\in\mathcal{N}^*}\phi_{k^{(1)}n}\cdot\ldots\cdot\phi_{k^{(r)}n} && \text{(A.18)}
\end{aligned}$$

Symbols $\mathcal{N}^*_{k^{(1)}},...,\mathcal{N}^*_{k^{(r)}}$ denote indexes of the top $@n$ mid-level features resulting from sorting by visual word $k^{(1)},...,k^{(r)}$, respectively. $\mathcal{N}^*$ denotes indexes of the top $@n$ mid-level features resulting from sorting by the joint occurrence of visual words $k^{(1)},...,k^{(r)}$. Note that $0\leq\phi_{kn}\leq 1$. Moreover, the above inequality holds as one can always find $\phi_{k^{(1)}n^{(1)}}\cdot\ldots\cdot\phi_{k^{(r)}n^{(r)}}$ that is greater than $\phi_{k^{(1)}n}\cdot\ldots\cdot\phi_{k^{(r)}n}$ in the following inequality:

$$\sum_{n^{(1)}\in\mathcal{N}^*_{k^{(1)}}}\phi_{k^{(1)}n^{(1)}}\cdot\ldots\cdot\sum_{n^{(r)}\in\mathcal{N}^*_{k^{(r)}}}\phi_{k^{(r)}n^{(r)}}\geq\sum_{n\in\mathcal{N}^*}\phi_{k^{(1)}n}\cdot\ldots\cdot\phi_{k^{(r)}n} \qquad \text{(A.19)}$$

Combining MaxExp, AxMin, or Gamma with the $@n$ operator preserves a somewhat similar bound because MaxExp, AxMin, or Gamma are non-decreasing functions $\upsilon(t):\langle 0;\infty\rangle\rightarrow\langle 0;\infty\rangle$ such that $\upsilon(t_2)\geq\upsilon(t_1)$ if $t_2\geq t_1$ and $\upsilon(t)\geq t$ for $0\leq t\leq 1$. Therefore:

$$\begin{aligned}
&\upsilon\left(\operatorname{avg}\operatorname{srt}\left(\left\{\phi_{k^{(1)}n}\right\}_{n\in\mathcal{N}}\right)\right)\cdot\ldots\cdot\upsilon\left(\operatorname{avg}\operatorname{srt}\left(\left\{\phi_{k^{(r)}n}\right\}_{n\in\mathcal{N}}\right)\right) && \text{(A.20)}\\
&\geq\upsilon^r\left(\frac{1}{@n^{1-\frac{1}{r}}}\operatorname{avg}\operatorname{srt}^{\frac{1}{r}}(\{\phi_{k^{(1)}n}\cdot\ldots\cdot\phi_{k^{(r)}n}\}_{n\in\mathcal{N}},@n)\right) && \text{(A.21)}
\end{aligned}$$

Note that $@n$ and r are typically low value constants, *e.g.* $@n=7$ and $r\leq 3$.

A.4 Statistical Significance

The paired t test has been performed to confirm that there is a statistical significance between Max-pooling and the proposed @n operator, presented in chapter 6, as well as between FK and Second-order Occurrence Pooling from chapter 7.

Figure 6.7. First, we demonstrate that MaxExp, AxMin, and Gamma operators are not statistically different. The theoretical similarity of these operators is argued in section 6.4.2. For MaxExp and Gamma groups resulting in 57.5±0.7 and 58.5±0.7% accuracy, the two-tailed P value equals 0.0538. By conventional criteria, this difference is considered to be not quite statistically significant. Groups, MaxExp and AxMin scored 57.5±0.7 and 57.5±0.5% accuracy. This difference is not statistically significant.

Table 6.3. For Max-pooling and AxMin@n (SC, SPM, 30 images/class) groups resulting in 80.4±0.6 and 81.3±0.6% accuracy, the two-tailed P value equals 0.045. By conventional criteria, this difference is considered to be statistically significant. For Max-pooling and AxMin@n (SC, SCC, 30 images/class) groups resulting in 68.0±0.5 and 71.6±0.4% accuracy, the two-tailed P value is $\leq$0.0001. By conventional criteria, this difference is considered to be extremely statistically significant. Also the results for Max-pooling and AxMin@n groups given LcSA are statistically significantly different.

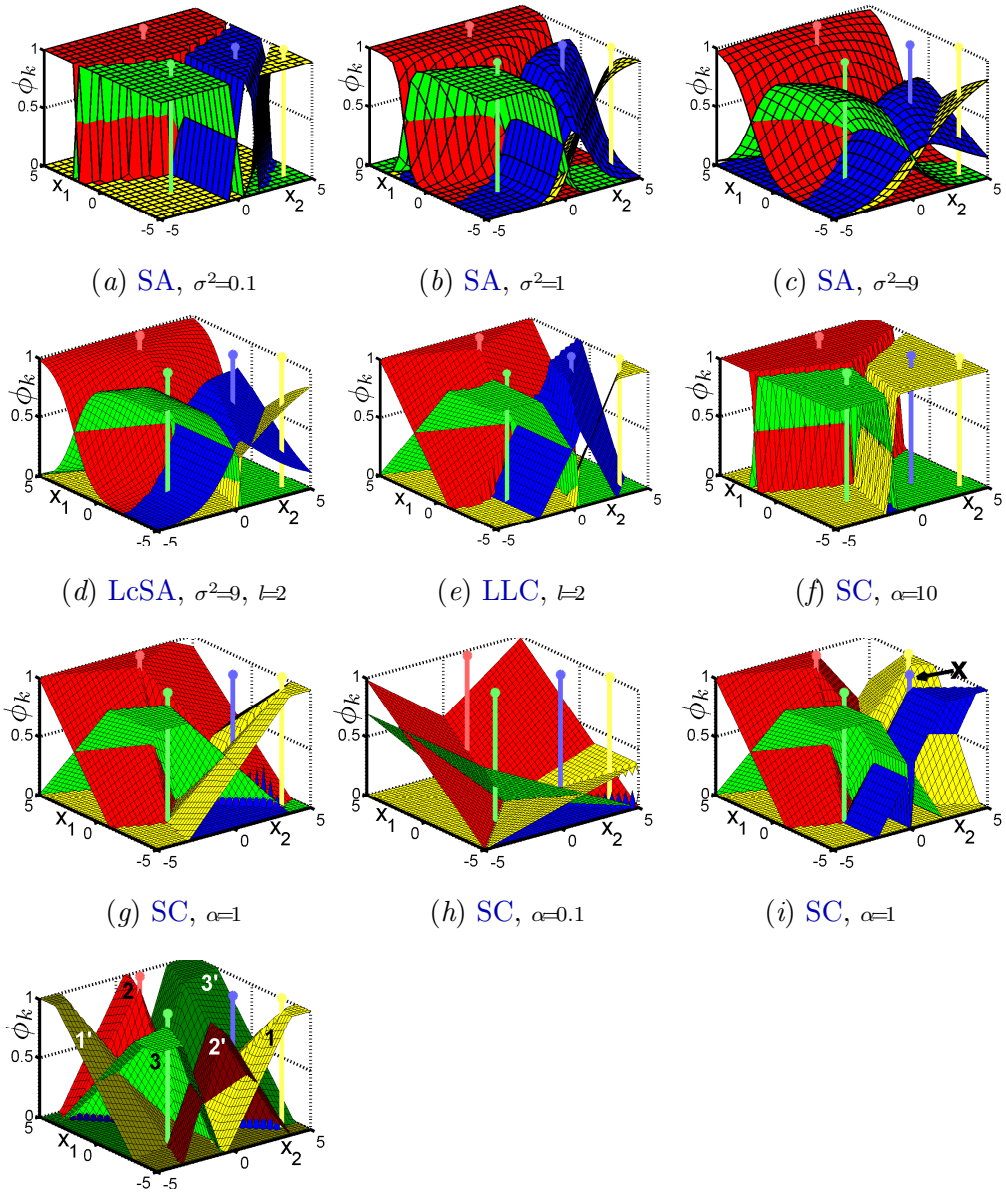
Table 6.5. For Max-pooling and AxMin@n groups given SC, LLC, and LcSA resulting in 93.4±0.3 and 94.4±0.4, 89.4±1.6 and 92.8±0.8, and 90.0±0.2 and 93.3±0.5% accuracy, respectively, the two-tailed P values are 0.0257, 0.0302, and 0.0004. Therefore, the differences in results given SC and LLC are both statistically significant. The difference given LcSA is considered to be extremely statistically significant.

Table 7.2. For FK and Second-order Occurrence Pooling (uni-modal $r = 2$) groups given 15 images/class, the results are 75.7±0.5 and 76.6±0.5% accuracy. In case of 30 images/class, these groups score 82.2±0.4 and 83.6±0.4% accuracy. The two-tailed P values are 0.0216 and 0.0006 given 15 and 30 images/class. By conventional criteria, these differences are considered to be statistically significant and extremely statistically significant, respectively.

For convenience, the t test calculator from [GraphPad, 2013] was used.

A.5 Activation Space of Various Coders

To introduce the SA, LcSA, LLC, and SC coding approaches better, we illustrate how they are affected by the coding parameters. Plots A.1 (a-c) present SA membership probabilities forming multidimensional activation functions spanned around four arbitrarily chosen anchors. Depending on σ, plot (a) shows SA acting as HA while plot (c) has locally linearised activation slopes. Plot (d) presents the probabilities of LcSA spanned locally between $l=2$ nearest neighbours of any given descriptor. The slopes are further linearised and appear similar to LLC shown in plot (e). Note, LcSA and LLC (unlike SA) have no overlapping activations $\phi_k \neq 0$ for descriptors that are not neighbours, *e.g.* $\boldsymbol{x}=[5,-5]^T$ and $\boldsymbol{x}=[-5,5]^T$, this depends on parameter l. Plots in figure A.1 (f-h) illustrate Sparse Coding activations (we rescaled these plots to $\langle 0;1\rangle$ range). Plot (f) shows that SC appears to act as HA for large α (however, true magnitude $\|\boldsymbol{\phi}\|_1=1/\alpha$ in this case). Plot (g) shows a case for moderate α. Plot (h) shows that for low α the largest ϕ_k are yielded for $\boldsymbol{x}$ situated far from anchors $\boldsymbol{m}_k$. Plot (i) shows that increasing the ℓ_2 norm of the anchor denoted by 'x' decreases its ℓ_1 norm regularisation cost and enables its corresponding activations (a new slope). SC suppresses any ϕ_k that pay a large ℓ_1 norm cost. Plot (j) shows negative activations of SC (we reversed the sign to be positive) denoted as 1', 2', and 3'. Vectors $\boldsymbol{x}$ inducing the negative values of these activations are far from the corresponding anchors 1, 2, and 3.



(*a*) SA, σ^2=0.1 (*b*) SA, σ^2=1 (*c*) SA, σ^2=9

(*d*) LcSA, σ^2=9, l=2 (*e*) LLC, l=2 (*f*) SC, α=10

(*g*) SC, α=1 (*h*) SC, α=0.1 (*i*) SC, α=1

(*j*) SC, α=1

Figure A.1: Activations ϕ_k for arbitrarily chosen $k = 1, ..., 4$ anchors $\boldsymbol{m}_k \in \langle -5; 5 \rangle^2$ and descriptors $\boldsymbol{x} = [x_1, x_2]^T \in \langle -5; 5 \rangle^2$. Membership probabilities given by (*a-c*) SA in equation (4.4) for various smoothing factors σ. (*d*) LcSA probabilities according to equation (6.9) for $l = 2$ nearest neighbours. (*e*) LLC activations according to equation (6.8). Activations given by (*f-h*) Sparse Coding with sparsity varied by α and responses rescaled to $\langle 0; 1 \rangle$ range. (*i*) Enabling activations of the anchor marked as 'x' by increasing its ℓ_2 norm. (*j*) Sparse Coding with dropped non-negativity constraint. Anchors (1-3) induce positive (1-3) and negative (1',2',3') activations. Best viewed in colour.

Bibliography

M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning. *Automation and Remote Control*, 25:821–837, 1964.

P. Arbelaez, C. Fowlkes, and D. Martin. The Berkley Segmentation Dataset and Benchmark, 2007. http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench.

S. Avila, N. Thome, M. Cord, E. Valle, and A.A. de Arajo. Pooling in Image Representation: The Visual Codeword Point of View. *CVIU*, 2012.

M. Awais, F. Yan, K. Mikolajczyk, and J. Kittler. Augmented Kernel Matrix vs Classifier Fusion for Object Recognition. *BMVC*, 2011a.

M. Awais, F. Yan, K. Mikolajczyk, and J. Kittler. Novel Fusion Methods for Pattern Recognition. *ECML*, 2011b.

M. Barnard, P. Koniusz, W. Wang, J. Kittler, S. M. Naqvi, and J. Chambers. A Robust Approach to Joint Audio-Visual Tracking Based on Bags of Visual Words. *TMM*, 2013. (submitted).

H. Bay, A. Ess, T. Tuytelaars, and L. van Gool. Speeded-up Robust Features (SURF). *CVIU*, 110(3):346–359, 2008. ISSN 10773142. doi: 10.1016/j.cviu.2007.09.014.

BBC Press Office. Roly Keating appointed as Director of Archive Content. www.bbc.co.uk/pressoffice/pressreleases/stories/2008/07_july/22/archive.shtml, 2008.

A. C. Berg and J. Malik. Geometric Blur for Template Matching. *CVPR*, pages 607–614, 2001.

I. Biederman. Recognition-by-components: A Theory of Human Image Understanding. *Psychological Review*, 94:115–147, 1987.

J. Bilmes. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report TR-97-021, ICSI, 1997.

A. Binder, W. Samek, and M. Kawanabe. The joint Submission of the TU Berlin and Fraunhofer FIRST (TUBFI) to the ImageCLEF 2011 Photo Annotation Task: Working Notes. *CLEF*, 2011.

A. Bosch, A. Zisserman, and X. Munoz. Image Classification using Random Forests and Ferns. *ICCV*, 2007.

S. Boughorbel, J-P. Tarel, and N. Boujemaa. Generalized Histogram Intersection Kernel for Image Recognition. *ICIP*, pages 161–164, 2005.

Y. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning Mid-Level Features for Recognition. *CVPR*, 2010a.

Y-L. Boureau, J. Ponce, and Y. LeCun. A Theoretical Analysis of Feature Pooling in Vision Algorithms. *ICML*, 2010b.

Y-L. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the Locals: Multi-way Local Pooling for Image Recognition. *ICCV*, 2011.

R. Bowden, J. Collomosse, and K. Mikolajczyk. Electronic Proceedings of the British Machine Vision Conference 2012. http://www.bmva.org/bmvc/2012, 2012.

L. Breiman. Random Forests. *ML*, 45(1):5–32, 2001. ISSN 0885-6125. doi: 10.1023/A:1010933404324.

L. Bretzner and T. Lindeberg. Feature Tracking with Automatic Selection of Spatial Scales. *CVIU*, 71:385–392, 1996.

D. Cai, X. He, and J. Han. Efficient Kernel Discriminant Analysis via Spectral Regression. *ICDM*, 2007.

M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary Robust Independent Elementary Features. *ECCV*, September 2010.

J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic Segmentation with Second-Order Pooling. *ECCV*, 2012.

C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using Expectation-Maximization and its application to image querying. *PAMI*, 24: 1026–1038, 1999.

C. C. Chang and C. J. Lin. LIBSVM: A Library for Support Vector Machines. *Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The Devil is in the Details: An Evaluation of Recent Feature Encoding Methods. *BMVC*, 2011.

A. Coates and A. Ng. The Importance of Encoding Versus Training with Sparse Coding and Vector Quantization. *ICML*, pages 921–928, June 2011.

D. Comaniciu and P. Meer. Mean Shift: A Robust Approach Toward Feature Space Analysis. *PAMI*, 24(5):603–619, 2002.

D. Comaniciu and P. Meer. Code for the Edge Detection and Image SegmentatiON system, 2003. http://www.caip.rutgers.edu/riul/research/code/EDISON/index.html.

C. Cortes and V. Vapnik. Support-Vector Networks. *ML*, 20(3):273–297, 1995.

T. Cour, S. Yu, and J. Shi. MATLAB Normalized Cuts Segmentation Code, 2004. http://www.cis.upenn.edu/~jshi/software.

G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual Categorization with Bags of Keypoints. *ECCV Workshop on Statistical Learning in Computer Vision*, pages 1–22, 2004.

N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. *CVPR*, 2:886–893, 2005.

P. Domingos and M. Pazzani. On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. *ML*, 29(2-3):103–130, 1997. ISSN 0885-6125. doi: 10.1023/A:1007413511361.

O. Duchenne, A. Joulin, and J. Ponce. A Graph-Matching Kernel for Object Categorization. *ICCV*, 2011.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley, 2 edition, 2001.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least Angle Regression. *Annals of Statistics*, 32:407–499, 2004.

F. J. Estrada and A. D. Jepson. Quantitative Evaluation of a Novel Image Segmentation Algorithm. *CVPR*, 2:20–26, 2005.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://pascallin.ecs.soton.ac.uk/challenges/VOC, 2007.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results. http://pascallin.ecs.soton.ac.uk/challenges/VOC, 2008.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. http://pascallin.ecs.soton.ac.uk/challenges/VOC, 2010.

L. Fei-fei, R. Fergus, and P. Perona. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. *CVPR Workshop on Generative-Model Based Vision*, 2004.

P. F. Felzenszwalb and D. P. Huttenlocher. Efficient Graph-Based Image Segmentation. *IJCV*, 59(2):167–181, 2005.

R. A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7(2):179–188, 1936.

D. Gabor. Theory of Communication. *J. IEE*, 93(26):429–457, 1946.

S. Gao, I. W. Tsang, L. Chia, and P. Zhao. Local Features Are Not Lonely - Laplacian Sparse Coding for Image Classification. *CVPR*, 2010.

F. Ge, S. Wang, and T. Liu. Image-Segmentation Evaluation From the Perspective of Salient Object Extraction. *CVPR*, 1:1146–1153, 2006.

Y. Gong, T. Huang, F. Lv, J. Wang, C. Wu, W. Xu, J. Yang, K. Yu, and T. Zhang. Image Classification Using Gaussian Mixture and Local Coordinate Coding. http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2009, 2009.

GraphPad. A t test calculator. http://www.graphpad.com/quickcalcs/ttest1.cfm, 2013.

K. Grauman and T. Darrell. The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features. *ICCV*, pages 1458–1465, 2005.

R. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-3(6):610–621, 1973.

C. Harris and M. Stephens. A Combined Corner and Edge Detector. *Alvey Vision Conference*, pages 147–151, 1988.

G. Haszprunar. The Mollusca: Coelomate Turbellarians or Mesenchymate Annelids? *Taylor, Origin and Evolutionary Radiation of the Mollusca*, pages 1–28, 1999.

V. Hedau, H. Arora, and N. Ahuja. Matching Images Under Unstable Segmentations. *CVPR*, 2008.

A. Hegerath, T. Deselaers, and H. Ney. Patch-based Object Recognition Using Discriminatively Trained Gaussian Mixtures. *BMVC*, 2:519–528, 2006.

A. Heyden and K. Rohr. Evaluation of Corner Extraction Schemes Using Invariance Methods. *ICPR*, 1:895–899, 1996.

M. J. Huiskes and M. S. Lew. The MIR Flickr Retrieval Evaluation. *MIR*, pages 39–43, 2008.

M. J. Huiskes, B. Thomee, and M. S. Lew. New Trends and Ideas in Visual Concept Detection: The MIR Flickr Retrieval Evaluation Initiative. *MIR*, pages 527–536, 2010.

L. Ibanez, W. Schroeder, L. Ng, and J. Cates. ITK Software Guide, 2005. http://www.itk.org/ItkSoftwareGuide.pdf.

ImageCLEF. Visual Concept Detection and Annotation Task 2011. http://www.imageclef.org/2011/Photo, 2011.

INDECT. Intelligent Information System Supporting Observation, Searching and Detection for Security of Citizens in Urban Environment. http://www.indect-project.eu, 2009.

T. Jebara, R. Kondor, and A. Howard. Probability Product Kernels. *JMLR*, 5:819–844, 2004. ISSN 1532-4435.

H. Jégou, M. Douze, and C. Schmid. On the Burstiness of Visual Elements. *CVPR*, pages 1169–1176, 2009.

H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating Local Descriptors into a Compact Image Representation. *CVPR*, pages 3304–3311, 2010.

H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P.Pérez, and C. Schmid. Aggregating Local Image Descriptors into Compact Codes. *PAMI*, 2012.

P. Jung. How Mobile is Changing the Way People Buy Motors. http://www.guardian.co.uk/media-network-partner-zone-ebay/ebay-mobile-consumers-vehicles-cars-motors?newsfeed=true, 2012.

P. Koniusz and K. Mikolajczyk. Segmentation Based Interest Points and Evaluation of Unsupervised Image Segmentation Methods. *BMVC*, 2009.

P. Koniusz and K. Mikolajczyk. On a Quest for Image Descriptors Based on Unsupervised Segmentation Maps. *ICPR*, 0:762–765, 2010. ISSN 1051-4651.

P. Koniusz and K. Mikolajczyk. Soft Assignment of Visual Words as Linear Coordinate Coding and Optimisation of its Reconstruction Error. *ICIP*, 2011a.

P. Koniusz and K. Mikolajczyk. Spatial Coordinate Coding to Reduce Histogram Representations, Dominant Angle and Colour Pyramid Match. *ICIP*, 2011b.

P. Koniusz, D. Bland, and K. Mikolajczyk. Classification and Retrieval of Images II. http://claret.wikidot.com, 2009.

P. Koniusz, F. Yan, and K. Mikolajczyk. Comparison of Mid-Level Feature Coding Approaches And Pooling Strategies in Visual Concept Detection. *CVIU*, 2012. ISSN 1077-3142. doi: 10.1016/j.cviu.2012.10.010.

P. Koniusz, F. Yan, P. Gosselin, and K. Mikolajczyk. Higher-order Occurrence Pooling on Mid- and Low-level Features: Visual Concept Detection. *PAMI*, 2013. (submitted).

N. Kulkarni and B. Li. Discriminative Affine Sparse Codes for Image Classification. *CVPR*, pages 1609–1616, 2011.

S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. *CVPR*, 2:2169–2178, 2006. ISSN 1063-6919.

H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient Sparse Coding Algorithms. *NIPS*, pages 801–808, 2007.

L. Lingqiao, L. Wang, and X. Liu. In Defence of Soft-assignment Coding. *ICCV*, 2011.

J. Liu, C. Zhang, Q. Tian, C. Xu, H. Lu, and S. Ma. One Step Beyond Bags of Features: Visual Categorization using Components. *ICIP*, 2011.

T. Liu, A. W. Moore, A. Gray, and K. Yang. An Investigation of Practical Approximate Nearest Neighbor Algorithms. *NIPS*, pages 825–832, 2004.

A. M. Lopez, F. Lumbreras, and J. J. Villanueva J. Serrat. Evaluation of Methods for Ridge and Valley Detection. *PAMI*, 21(4):327–335, 1999.

D. G. Lowe. Object Recognition from Local Scale-Invariant Features. *CVPR*, 2:1150–1157, 1999.

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online Learning for Matrix Factorization and Sparse Coding. *JMLR*, 2010.

T. Malisiewicz and A. Efros. Improving Spatial Support for Objects via Multiple Segmentations. *BMVC*, 2007.

S. Mallat and Z. Zhang. Matching Pursuit with Time-Frequency Dictionaries. *TSP*, 41:3397–3415, 1993.

M. Marszalek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning Object Representations for Visual Object Class Recognition. *ICCV Workshop on The PASCAL VOC07 Challenge*, 2007.

D. Martin, C. Fowlkes, D. Tal, and J. Malik. A Database of Human Segmented Natural Images and its Application to Evaluation Segmentation Algorithms and Measuring Ecological Statistics. *ICCV*, 2:416–423, 2001.

J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. *BMVC*, pages 384–393, 2002.

S. Mika, G. Ratsch, J. Weston, B. Scholkoph, and K. Mullers. Fisher Discriminant Analysis with Kernels. *Neural Networks for Signal Processing*, 1999.

K. Mikolajczyk and C. Schmid. A Performance Evaluation of Local Descriptors. *PAMI*, 27(10):1615–1630, 2005.

K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. van Goll. A comparison of affine region detectors. *IJCV*, 65(1-2): 43–72, 2005.

H. Mller and J. Kalpathy-Cramer. The ImageCLEF Medical Retrieval Task at ICPR 2010 - Information Fusion. *ICPR*, pages 3284–3287, 2010.

F. Mokhatarian, S. Abbasi, and J. Kittler. Efficient and Robust Retrieval by Shape Content through Curvature Scale Space. *International Workshop on Image Databases and Multimedia Search*, pages 35–42, 1996.

MOSEK. The MOSEK Optimization Software, 2012.

R. Negrel, D. Picard, and P-H. Gosselin. Compact Tensor Based Image Representation for Similarity Search. *ICIP*, 2012.

M. E. Nilsback and A. Zisserman. A Visual Vocabulary for Flower Classification. *CVPR*, 2:1447–1454, 2006.

M. E. Nilsback and A. Zisserman. Automated Flower Classification over a Large Number of Classes. *ICVGIP*, Dec 2008a.

M. E. Nilsback and A. Zisserman. Automated Flower Classification over a Large Number of Classes. *ICVGIP*, pages 722–729, 2008b.

E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. *ECCV*, pages 490–503, 2006.

S. Nowak, K. Nagel, and J. Liebetra. The CLEF 2011 Photo Annotation and Concept-based Retrieval Tasks. *CLEF*, 2011.

T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns. *PAMI*, 24(7):971–987, 2002.

P. Ott and M.Everingham. Implicit Color Segmentation Features for Pedestrian Detection. *ICCV*, 2009.

P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *PAMI*, 12:629–639, 1990.

F. Perronnin and C. Dance. Fisher Kernels on Visual Vocabularies for Image Categorization. *CVPR*, 0:1–8, 2007.

F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher Kernel for Large-Scale Image Classification. *ECCV*, pages 143–156, 2010.

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases. *CVPR*, 2008.

David Picard and Philippe-Henri Gosselin. Improving Image Similarity with Vectors of Locally Aggregated Tensors. *ICIP*, 2011.

J. R. Quinlan. Induction of Decision Trees. *ML*, 1(1):81–106, 1986. ISSN 0885-6125. doi: 10.1023/A:1022643204877.

R. C. Rao. The Utilization of Multiple Measurements in Problems of Biological Classification. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 1948.

S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. *CVPR*, pages 860–867, 2005.

B. C. Russel, A. A. Efros, J. Sivic, W. T. Freeman, and A. Zisserman. Using Multiple Segmentations to Discover Objects and their Extent in Image Collections. *CVPR*, pages 1605–1614, 2006.

J. Sánchez, F. Perronnin, and T. E. de Campos. Modeling the Spatial Layout of Images Beyond Spatial Pyramids. *PRL*, 2012.

R. E. Schapire. The Strength of Weak Learnability. *ML*, 5(2):197–227, 1990. ISSN 0885-6125. doi: 10.1023/A:1022648800760.

C. Schmid, R. Mohr, and C. Buackhage. Evaluation of Interest Point Detectors. *IJCV*, 37(2):151–172, 2000.

B. Scholkopf, K. Sung, C. J. C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik. Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers. *SP*, 45(11):2758–2765, November 1997. ISSN 1053-587X. doi: 10.1109/78.650102.

J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2002.

C. Siagian and L. Itti. Rapid Biologically-Inspired Scene Classification Using Features Shared with Visual Attention. *PAMI*, 29(2):300–312, 2007.

J. Sivic and A. Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. *ICCV*, 2:1470–1477, 2003.

M. Smith and J. M. Brady. Susan-a new approach to low level image processing. *IJCV*, 23(1):45–78, 1997.

D. Song and T. Dacheng. Biologically Inspired Feature Manifold for Scene Classification. *TIP*, 19(1):174–184, 2010.

J. Stöttinger, A. Hanbury, N. Sebe, and T. Gevers. Sparse Color Interest Points for Image Retrieval and Object Categorization. *TIP*, 2012.

Y. Su and F. Jurie. Semantic Contexts and Fisher Vectors for the ImageCLEF 2011 Photo Annotation Task. *CLEF*, 2011.

M. A. Tahir, J. Kittler, K. Mikolajczyk, F. Yan, K. van de Sande, and T. Gevers. Visual Category Recognition using Spectral Regression and Kernel Discriminant Analysis. *ICCV Workshop on Subspace Methods*, 2009.

M. A. Tahir, F. Yan, M. Barnard, M. Awais, K. Mikolajczyk, and J. Kittler. The University of Surrey Visual Concept Detection System at ImageCLEF 2010: Working Notes. *ICPR*, 2010.

M. A. Tahir, F. Yan, P. Koniusz, M. Awais, M. Barnard, K. Mikolajczyk, and J. Kittler. A Robust and Scalable Visual Category and Action Recognition System using Kernel Discriminant Analysis with Spectral Regression. *TMM*, 2012.

E. Tola, V. Lepetit, and P. Fua. A Fast Local Descriptor for Dense Matching. *CVPR*, 2008.

T. Tommasi and T. Deselaers. The Medical Image Classification Task. *The Information Retrieval Series*, 32:221–238, 2010.

A. Torralba, R. Fergus, and W. T. Freeman. 80 Million Tiny Images: A Large Data Set for Nonparametric Object and Scene Recognition. *PAMI*, 30(11):1958–1970, 2008. ISSN 0162-8828.

I. Tosic and P. Frossard. Dictionary Learning. *SPM*, 28(2):27–38, 2011.

J. A. Tropp. Greed is Good: Algorithmic Results for Sparse Approximation. *TIT*, 50: 2231–2242, 2004.

T. Tuytelaars and K. Mikolajczyk. A Survey on Local Invariant Features. *Foundations and Trends in Computer Graphics and Vision*, 3:177–280, 2008.

T. Tuytelaars and C. Schmid. Vector Quantizing Feature Space with a Regular Lattice. *ICCV*, 2007.

K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. A Comparison of Color Features for Visual Concept Classification. *CIVR*, pages 141–149, 2008.

J. C. van Gemert, J. M. Geusebroek, C. Veenman, and A. Smeulders. Kernel Codebooks for Scene Categorization. *ECCV*, 5304:696–709, 2008.

J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek. Visual Word Ambiguity. *PAMI*, 2010.

J. K. M. Vetterli. *Wavelets and Subband Coding*. Prentice Hall, 1995.

J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained Linear Coding for Image Classification. *CVPR*, 2010.

L. Wang. Toward A Discriminative Codebook: Codeword Selection across Multi-resolution. *CVPR*, 0:1–8, 2007.

J. L. Weed and J. M. Raber. Balancing Animal Research with Animal Well-being: Establishment of Goals and Harmonization of Approaches. *Institute for Laboratory Animal Research Journal*, 46:118–128, 2005.

S. A. J. Winder and M. Brown. Learning Local Image Descriptors. *CVPR*, 2007.

G. W. Witmer. Wildlife Population Monitoring: Some Practical Considerations. *Wildlife Research*, 32:259–263, 2005.

F. Yan, K. Mikolajczyk, M. Barnard, H. Cai, and J. Kittler. Lp Norm Multiple Kernel Fisher Discriminant Analysis for Object and Image Categorisation. *CVPR*, 2010.

J. Yang, Y-G. Jiang, A. G. Hauptmann, and C-W. Ngo. Evaluating Bag-of-Visual-Words Representations in Scene Classification. *MIR*, pages 197–206, 2007.

J. Yang, K. Yu, Y. Gong, and T. S. Huang. Linear Spatial Pyramid Matching using Sparse Coding for Image Classification. *CVPR*, pages 1794–1801, 2009.

J. Yang, K. Yu, and T. Huang. Efficient Highly Over-Complete Sparse Coding using a Mixture Model. *ECCV*, pages 113–126, 2010.

J. Yang, Y. Tian, L. Duan, T. Huang, and W. Gao. Group-Sensitive Multiple Kernel Learning for Object Recognition. *TIP*, 21(5):2838–2852, 2012a.

J. Yang, Z. Wang, Z. Lin, X. Shu, and T. Huang. Bilevel Sparse Coding for Coupled Feature Spaces. *CVPR*, 2012b.

K. Yu, T. Zhang, and Y. Gong. Nonlinear Learning using Local Coordinate Coding. *NIPS*, 2009.

Xinnan Yu and Y-J. Zhang. A 2-D Histogram Representation of Images for Pooling. *SPIE*, 2011.

X-T. Yuan and S. Yan. Visual Classification with Multi-Task Joint Sparse Representation. *CVPR*, 2010.

C. Zhang, Q. Huang, J. Liu, Q. Tian, C. Liang, and X. Zhu. Image Classification Using Haar-like Transformation of Local Features with Coding Residuals. *SP*, 2012.

H. Zhang, J. E. Fritts, and S. A. Goldman. Image Segmentation Evaluation: A Survey of Unsupervised Methods. *CVIU*, 2(110):260–280, 2008.

X. Zhou, K. Yu, T. Zhang, and T. S. Huang. Image Classification using Super-Vector Coding of Local Image Descriptors. *ECCV*, pages 141–154, 2010.

M. Zhu. Recall, Precision and Average Precision. http://sas.uwaterloo.ca/stats_navigation/techreports/04WorkingPapers/2004-09.pdf, 2004.

Zooniverse. Planet Four: A Citizen Science Project. http://planetfour.org, 2012.