Content-based feature weighting for scene recognition

Zhu, Jing

2010

Zhu, J. (2010). Content-based feature weighting for scene recognition. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/38771>

<https://doi.org/10.32657/10356/38771>

Content-based Feature Weighting for Scene Recognition

Zhu Jing

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Master of Engineering

2010

Acknowledgments

I would like to thank first and foremost my supervisor, Prof. Kai-Kuang Ma, for his professional guidance, invaluable assistance, and kindly encouragement throughout my M.Eng study as well as his valuable advices for the completion of this thesis. At every stage of my work, his insights and his incisive questioning have been crucial in bringing my efforts to fruition. He always gave me strong support and sincere encouragement throughout the tough times of both the research and the life, particularly when I got sick and had to be in hospital in the first year.

I also wish to take this opportunity to thank School of EEE, Nanyang Technological University for offering research facilities for conducting research. I am very grateful to a number of people who directly or indirectly influenced the work in this thesis: Tian Jing, Wei Zhe, Wang Chen, Chai Zhi, Chen Yibin and Tang Tianyu. It is also my great pleasure to thank and acknowledge my colleagues in Media Technology Lab and Temasek Lab at NTU, for creating a friendly working environment.

Lastly but most important, I wish to dedicate this thesis to my family, whose love and support are always the greatest inspiration for me in my life. Thank my parents, for their care, understanding and encouragement during my entire M.Eng study.

Contents

Acknowledgments	i
List of Figures	iv
List of Tables	viii
Summary	ix
1 Introduction	1
1.1 Motivation	1
1.2 Contributions	4
1.3 Organization of the Thesis	6
2 Background	8
2.1 The Bag of Features (BoF) Model Based Image Representation . .	8
2.1.1 Feature Detection	10
2.1.2 Feature Description	16
2.1.3 Clustering and Visual Vocabulary Formulation	20
2.1.4 Visual Words Assignment	23
2.1.5 Discussion	26
2.2 Brin and Page's PageRank Algorithm	28

3	FeatureRank Weighting and Its Use for the BoF Model	31
3.1	Proposed FeatureRank Weighting and Its Applications	31
3.1.1	FeatureRank	31
3.1.2	Nonparametric FeatureRank-based Clustering	39
3.2	FeatureRank-based Weighting Scheme for the BoF Model	47
3.2.1	Detailed Weighting Scheme	47
3.2.2	Experiments	52
4	Conclusion and Future Work	59
4.1	Conclusion	59
4.2	Future Work	60
A	The Bag of Words (BoW) Model	64

List of Figures

1.1	Eiffel tower captured under different conditions: (a) a normal frontal view; (b) a night scene (thus, incurring a drastic illumination change); (c) a zoom-out view with a substantial change in object size; (d) with a further rotation; (e) with a substantial change on viewing angle; (f) with a partial occlusion.	3
1.2	Illustration of the proposed FeatureRank algorithm.	5
1.3	Illustration of the proposed BoF model with FeatureRank-based weighting scheme.	7
2.1	The flowchart of BoF	9
2.2	Local features detected by the Harris detector on rotated images. .	12
2.3	Corresponding local features detected by the Harris-Laplace detector. The circles indicate the scale of the features.	12
2.4	Local features detected by the Harris-Affine detector. The region shapes in two images with different viewpoints clearly correspond. .	13
2.5	Local features detected by the Hessian detector on rotated images .	13
2.6	Local features detected by the Hessian-Laplace detector on images with scale change	14
2.7	Local features detected by the Hessian-Affine detector from two different view	14

2.8	Local features detected by the DoG detector.	15
2.9	Local features detected by the MSER detector.	16
2.10	Illustration of the SIFT descriptor partially taken from [1] and [2]. After orientation assignment for interested points, a histogram of gradient orientations is formed in each grid cell. 2×2 grid is depicted for a better facility of illustration. 8 orientation bins are used in each grid cell therefore giving a descriptor of dimension $32 = (2 \times 2 \times 8)$	18
2.11	Illustration of the GLOH descriptor taken from [2].	20
2.12	Illustration of the process of building the vocabulary tree taken from [3]. The hierarchical quantization is defined at each level by k clusters (in this case $k = 3$).	21
2.13	The benefits of soft-assignment technology [4]	25
2.14	Illustration of the work of [5,6].	25
3.1	Illustration examples of FeatureRank on 2D regular-shaped synthetic data. left: 2D synthetic data ; right: 2D synthetic data weighted by FeatureRank (The different colors stand for different values).	36
3.2	Illustration examples of FeatureRank on 2D symmetrical structured data. left: 2D synthetic data ; right: 2D synthetic data weighted by FeatureRank (The different colors stand for different values).	37
3.3	Illustration examples of FeatureRank performance with different pa- rameter k (The different colors stand for different values).	38
3.4	Illustration of proposed nonparametric FeatureRank-based Clustering	42

3.5	Clustering algorithms comparisons on 2D regular-shaped synthetic dataset (one line cluster and two compact spherical clusters). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=1$) (d) result of proposed nonparametric FeatureRank-based clustering	43
3.6	Clustering algorithms comparisons on 2D regular-shaped synthetic dataset (two line cluster and one compact spherical clusters). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=10$) (d) result of proposed nonparametric FeatureRank-based clustering	44
3.7	Clustering algorithms comparisons on symmetrical structured datasets (ring-shaped cluster). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=1$) (d) result of proposed nonparametric FeatureRank-based clustering	45
3.8	Clustering algorithms comparisons on symmetrical structured datasets (ellipsoidal shell-shaped cluster). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=0.05$) (d) result of proposed nonparametric FeatureRank-based clustering	46
3.9	Text-based weighting scheme using in BoF which produces similarity scores equal to zero	49
3.10	The illustration of over-discriminative visual words	51
3.11	Database1: 4 natural scenes categories from [7], (a) inside of cities, (b) streets, (c) suburb residence, (d) tall buildings	53
3.12	Database2: Paris Database from [4], (a) La Defense, (b) Eiffel Tower, (c) Hotel des Invalides, (d) Louvre, (e) Moulin Rouge, (f) Arc de Triomphe, (g) Notre Dame, (h) Pantheon, (i) Sacre Coeur	54

3.13	The average precision curve comparisons of two weighting schemes: Proposed FeatureRank-based weighting and TF-IDF, on database of 4 natural scenes categories	57
3.14	The average precision curve comparisons of two weighting schemes: Proposed FeatureRank-based weighting and TF-IDF, on 9 location categories of paris dataset	57
3.15	The average precision comparisons of two weighting schemes: Pro- posed FeatureRank-based weighting and TF-IDF, on each location category of paris dataset (the vocabulary size=4000)	58

List of Tables

- 3.1 Text retrieval-based weighting schemes for the BoF model (t_i : the i th word; d_j : the j th document; w_{ij} : weight for t_i in document d_j ; tf_{ij} : the number of occurrences of word t_i in document d_j ; N : total number of documents; n_i : the number of documents having word t_i ; TF: term frequency; IDF: inverse document frequency.) 48
- 3.2 The average precision comparisons of two weighting schemes: Proposed FeatureRank-based weighting and TF-IDF, on two databases. 56

Summary

Image representation for content-based scene recognition is considered as one of the most challenging problems in computer vision. Motivated by the successful text information retrieval application through a histogram of word counts for each text document, the promising *bag of features* (BoF) model (or the *bag of visual words* model) has been developed in recent years and clearly demonstrated its potential as a powerful image representation technique for visual information retrieval application. The BoF model represents each image as a collection of clustered local features, and the centroid of each cluster is known as a *visual word*, which is analogous to a text word. Since the contextual information (i.e., the inter-word relationship) among the visual words is omitted in the conventional BoF model, a novel feature weighting method, coined as the *FeatureRank* (FR) in this thesis, has been proposed to address this issue. Furthermore, a non-parametric clustering scheme has been developed, which does not require any prior knowledge regarding the number of clusters involved. Finally, the FeatureRank-based weighting scheme has been developed and incorporated into the BoF model for improving the performance of scene recognition.

The proposed FeatureRank, which is used for weighting the features in the feature space, is mathematically derived based on a core thesis that is analogous to the Brin and Page's *PageRank* (PR), which is used for the ranking of web pages by popularity. Our core thesis says: *a feature is considered 'important' if it is similar*

to other features that have been identified ‘important’ already. Based on this, a contextual hyperlink relationship among the features is constructed by finding the similar neighbors of each feature. Consequently, the importance of each feature (i.e., its weight), can be iteratively computed by applying a Markov Chain process.

FeatureRank is a good indicator using in describing how dense the features are aggregated together, as it will assign the larger weight to the features in dense regions. In virtue of this property, a non-parametric FeatureRank-based clustering algorithm without requiring the number of clusters has also been developed in this thesis. Simulation results based on multiple two-dimensional synthetic datasets verify that the proposed FeatureRank and the nonparametric FeatureRank-based clustering algorithm are all effective.

Finally, a BoF model with FeatureRank-based weighting scheme has been investigated. Unlike the traditional BoF model, which does not consider any semantic relationship existing among the visual words, the proposed FeatureRank-based weighting scheme, on the other hand, exploits FeatureRank to better distinguish each visual word through the use of contextual information existing among the visual words. Simulations based on two widely used databases were conducted, and the results show that the proposed weighting scheme have yielded attractive performance for scene recognition application.

Chapter 1

Introduction

1.1 Motivation

Content-based visual search for scene recognition has been finding its way in diverse fields of applications, such as image or video shots retrieval over personal photo databases [1–3] or the internet image databases [4, 7, 8], image-based location-aware mobile positioning systems [9], and automated surveillance systems [10], to name a few. One of the challenges that content-based visual search face is how to represent the image data according to their *semantic* content. This challenge is further compounded by the presence of illumination variations, rotation and scale changes, different scene acquisition viewpoints, and object occlusions with possible clutter in their background, as demonstrated in Fig. 1.1. To address them, numerous image representation approaches have been developed in the past [1–7, 11–13]. Early methods used object boundaries to represent image, which are appealing as they are invariant to a large amount of illumination changes. In practice, however, these methods suffer from not being robust enough to handle the images with partial occlusion and/or a large amount of background clutter [12, 13].

Recently, the *bag of features* (BoF) model [7, 11], which represents an image as

an orderless collection of *local* image features, has attracted wide attentions as this model shows some robustness under a broad class of imaging conditions. It particularly works well for handling the problem of partial occlusion and large background clutter [3–6]. The BoF is motivated by the *bag of words* (BoW) model [14] developed for the text information retrieval application, where each text document is represented as a histogram of word counts. Similarly, the key idea behind the BoF is to cluster or group the extracted high-dimensional local image features to a vocabulary of *visual words*, which is analogous to the words encountered in text. A set of visual words could correspond to a portion of an object or to the different parts of several objects.

The description of an image using the BoF model relies on two steps: 1) generating its visual words; and 2) representing it by using the generated visual words. The first step is an *off*-line training process, operated over a large training image database. More specifically, the local features of each image from the database are extracted by exploiting a feature detector and a feature descriptor, followed by grouping them according to a specified number of clusters using an unsupervised clustering algorithm (e.g., the *k*-means [15]). As a result, the centroids of clusters are treated as the *visual words*. Unlike the first step, the second step is an *on*-line process, performed at the query stage. Given a query image, by mapping its extracted local features (using the same feature detector and descriptor) against the visual words from the vocabulary, the best match can be found. Then the query image is represented by a histogram over the vocabulary of visual words.

One fundamental problem existing in the conventional BoF model is that it weights each visual word *equally* [7]. Intuitively speaking, this is *not* appropriate, since visual words should carry various degrees of importance as reflected by their semantic content. Thus, how to determine the weight of each visual word is an important issue to be studied. Since the BoF shares some common ground with

Figure 1.1. Eiffel tower captured under different conditions: (a) a normal frontal view; (b) a night scene (thus, incurring a drastic illumination change); (c) a zoom-out view with a substantial change in object size; (d) with a further rotation; (e) with a substantial change on viewing angle; (f) with a partial occlusion.

that of the traditional technique used in the text retrieval application [4, 16, 17], certain weighting schemes for text retrieval could be applied for image retrieval; for instance, the classical *term frequency-inverse document frequency* (TF-IDF) weighting method [11, 18]. Nevertheless, most of the weighting methods used for text retrieval rely on the assumption that each text word occurs independently; this is generally acceptable in text retrieval, but inappropriate in image retrieval, since several visual words are often needed to describe one image object. Therefore, for the BoF model, it is essential to take the *contextual* information (i.e., the inter-word relationship) among the visual words into account for conducting proper weighting operation.

1.2 Contributions

In this thesis, three issues were investigated.

First, a new feature weighting method called the *FeatureRank* (FR), which considered both the local and global similarity information among features in the weighting process, was exploited, as depicted in Fig. 1.2. *FeatureRank* (FR), applying the idea of ranking by popularity to the feature space for weighting, was established based on the Brin and Page’s ranking technique—*PageRank* (PR) [19]. Similar to the thesis assumption made by Brin and Page in their PR algorithm, our proposed FeatureRank ranks each feature in the feature space according to the following core thesis: *a feature is considered ‘important’ if it is similar to other features that have been identified ‘important’ already*. The proposed FeatureRank, firstly constructs a contextual hyperlink matrix by finding each feature’s similar neighbors to preserve the local information among the features. A Markov Chain model is then applied to an iterative computing in order to take global structural relationships among all features into account and conduct feature weighting.

Furthermore, to demonstrate the effectiveness of FeatureRank, a nonparametric clustering based on FeatureRank was then developed. Since FeatureRank will assign the larger weight to the features in dense regions, it is a good indicator using in describing how dense the features are aggregated together. Based on this property, this nonparametric FeatureRank-based clustering algorithm can produce clusters depending upon both the local and global structure of the features without requiring any priori knowledge regarding the number of clusters (as required in the k -means clustering algorithm [15]).

Finally, a BoF model with proposed FeatureRank-based weighting scheme was built up, as depicted in Fig. 1.3. In our BoF model, Harris-Affine detectors and

Figure 1.2. Illustration of the proposed FeatureRank algorithm.

SIFT descriptors, which has been widely used nowadays, are used in feature detection and feature description, respectively. The k -means clustering is performed to construct a visual vocabulary. After that, a new weighting scheme based on *FeatureRank* (FR) are conducted. Unlike the traditional text-based weighting schemes for the BoF model, which ignore the contextual information among visual words, the proposed weighting scheme first computes FeatureRank value of visual words to emphasize both the local and global similarity information for weighting and then generates the weighted histogram for the BoF model. With the aid of FeatureRank to incorporate the *contextual* information, the proposed weighting scheme addresses the issue of lacking inter-word relationship among visual words in the traditional text-based weighting schemes for the BoF model.

1.3 Organization of the Thesis

Chapter 2 reviews the fundamentals of the BoF model and the Brin and Page's PageRank algorithm, including its related mathematics. In Chapter 3, our proposed FeatureRank algorithm for weighting the visual words (i.e., local features) will be mathematically derived based on a core thesis, which is analogous to the ranking of web pages by popularity as initially introduced by Brin and Page in their PageRank (PR) algorithm. Moreover, a non-parametric clustering algorithm which does not require any a priori knowledge regarding the number of clusters imposed to be used is also proposed in this chapter. In addition, the FR-based weighting scheme for the BoF model was established to further address the problem of neglecting the inter-word relationship among the visual words encountered in the conventional BoF model. Finally, conclusions and future work are provided in Chapter 4.

Figure 1.3. Illustration of the proposed BoF model with FeatureRank-based weighting scheme.

Chapter 2

Background

This chapter provides a comprehensive survey of literature related to this thesis in two parts. Firstly, the fundamental principle of the *bag of features* (BoF) model are reviewed in details. Secondly, PageRank algorithm and its mathematic model which served as the background work of the proposed FeatureRank weighting method are presented.

2.1 The Bag of Features (BoF) Model Based Image Representation

Image representation for content-based scene recognition poses significant challenge for computer vision especially in the presence of a broad class of imaging conditions which typically include varying camera view point, scene lighting, object occlusion, possible object deformation and background clutter. Recently the *bag of features* (BoF) model [7, 11, 20–22] based on a set of local features has been proposed by Sivic and Zisserman [11] and has shown robustness to recognize objects or scenes in spite of large changes in viewing conditions, occlusions and image clutter [4, 7, 11, 16, 18, 21]. Analogous to the *bag of words* (BoW) [14] representation of text documents

Figure 2.1. The flowchart of BoF

in terms of form and semantics, this model represents an image as a histogram of quantized local features called *visual words* (which is analogous to words in text). In view of this, the detailed introduction of the *bag of words* (BoW) [14] representation of text documents are documented in the appendix of this thesis.

Two key issues should be addressed in the BoF model. The first is how to define the *visual word* (analogy of *word* in text document), and the second issue is how to use the visual words to represent an image. To achieve this, the BoF model usually includes following steps shown in Fig. 2.1, which consists of two stages: Firstly, an off-line training stage which groups the local features extracted by some feature detectors and descriptors from a large training image database into specified number of clusters using an unsupervised clustering algorithm such as k -means is performed and the centers of clusters are considered as visual words. Successively, an on-line query process is conducted by assigning each feature extracted by the same feature detectors and descriptors as those of training stage from a query image to its nearest visual word, and representing the query image by a histogram over the vocabulary of visual words. The following parts in this section will focus on introducing each steps mentioned above in details.

2.1.1 Feature Detection

Feature detection which extracts a set of local interest points and salient image regions serves as the first step and also forms the basis of the BoF model. The detected local features are expected to be invariant to geometric changes (e.g., various viewpoints) and photometric changes (e.g., different illumination conditions) so as to be robust in spite of various scales, illuminations, and viewpoint conditions. Numerous research on different feature detectors which emphasize different aspects of invariance, resulting in local features of varying properties and sampled size are reported in recent years [1, 23–32]. The most popular feature detectors which give sufficient performance results were shown in [25, 33, 34]. Those related with our work in this thesis are as follows: *Harris-based* detectors (Harris, Harris-Laplace, Harris-Affine) [26–28], *Hessian Matrix-based* detectors (Hessian, Hessian-Laplace, Hessian-Affine) [23], *Difference of Gaussian* detector (DoG) [1], *Maximally Stable Extremal Regions* detector (MSER) [29].

In this section, a brief introduction of these feature detectors will be given in turn:

1) *Harris-based* detectors

The Harris detector proposed in [27], as shown in Fig. 2.2, which responds to *corner-like* features, is the simplest one in the Harris-based detectors. It is based on the second moment matrix which describes the gradient distribution in a local neighborhood of a point, as follows,

$$\mu(\mathbf{x}) = \begin{bmatrix} I_x^2(\mathbf{x}) & I_x I_y(\mathbf{x}) \\ I_x I_y(\mathbf{x}) & I_y^2(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} A & B \\ B & C \end{bmatrix} \quad (2.1)$$

where $\mathbf{x} = (x, y)$, I_x and I_y are the respective derivatives (of pixel intensity) in the x and y direction. The corner response c is efficiently calculated by avoiding the eigenvalue decomposition of the second moment matrix by

$$c = \text{Det}(\mu(\mathbf{x})) - k \times \text{Tr}(\mu(\mathbf{x}))^2 = (AC - B^2) - k \times (A + C)^2 \quad (2.2)$$

where k is an empirically determined constant. This is followed by a non-maximum suppression step and then a Harris corner is identified by a high positive response of the cornerness measure c . The simple Harris detector delivers a large number of interest points with sufficient repeatability as shown in [34]. The main advantage of this detector is the speed of calculation and the rotational, illumination invariance properties. A disadvantage is the fact that the detector determines only the spatial locations of the interest points. No region properties such as scale or orientation are determined for the consecutive descriptor calculation.

Thus the Harris-Laplace detector [28] is exploited with the idea of incorporating scale invariance property into the detector, as shown in Fig. 2.3. It uses a scale-adapted Harris function to localize points in scale-space, and then selects the points for which the *Laplacian of Gaussian* (LoG) attains a maximum over a scale. However, the Harris-Laplace satisfying scale invariant just work well for the isotropic regions that are viewed from the same viewing angle. In order to obtain invariance against affine transformed images with different viewpoints, based on the work of Harris-Laplace detector, Harris-Affine detector is proposed by Mikolajczyk and Schmid [23] which estimates the affine neighborhood by the affine adaptation based on the second moment matrix. Fig. 2.4 shows that, in spite of the affine deformation caused by two different viewpoints, the regions detected by the Harris-Affine detector clearly correspond.

2) *Hessian Matrix-based* detectors

The Hessian Matrix-based detectors [23] are based on a similar idea like Harris-based corner detectors, however, they respond to *blob-like* features. They are in

Figure 2.2. Local features detected by the Harris detector on rotated images.

Figure 2.3. Corresponding local features detected by the Harris-Laplace detector. The circles indicate the scale of the features.

principle based on the Hessian matrix defined as:

$$H(\mathbf{x}) = \begin{bmatrix} I_{xx}(\mathbf{x}) & I_{xy}(\mathbf{x}) \\ I_{xy}(\mathbf{x}) & I_{yy}(\mathbf{x}) \end{bmatrix} \quad (2.3)$$

where $\mathbf{x} = (x, y)$, I_{xx} and I_{yy} are the second derivatives; I_{xy} is the mixed derivative, of pixel intensity I in the x and y direction, respectively. The selection criterion for Hessian points is based on the determinant of the Hessian-matrix after non-maximum suppression. The simplest Hessian detector only shows rotational and illumination invariance properties, as shown in Fig. 2.5.

Same as the scale adaptations of Harris-based detectors, in Hessian's case, the local extremum of the scale normalized Laplacian S is used as a scale selection criterion to yield a scale invariant Hessian-Laplace detector [23], as shown in Fig.

Figure 2.4. Local features detected by the Harris-Affine detector. The region shapes in two images with different viewpoints clearly correspond.

Figure 2.5. Local features detected by the Hessian detector on rotated images

2.6, see the following equation. The standard deviation of Gaussian smoothing for scale space generation (often also termed local scale) is denoted by s :

$$S = s^2 \times |(I_{xx}(\mathbf{x}) + I_{yy}(\mathbf{x}))| \quad (2.4)$$

Apart from scale invariance, such Hessian-Laplace detector also satisfies the good rotational and illumination invariance. Furthermore, the Hessian-Affine detector [23] which serves as a viewpoint invariant blob detector is achieved after the affine adaptation procedure based on Hessian Laplace detector, as shown in Fig. 2.7.

3) *Difference of Gaussian* detector (DoG)

The Difference of Gaussian detector [1, 35], or DoG for short, is a scale invariant detector which extracts *blob-like* features in the image by calculating differences of Gaussian blurred images at several adjacent local scales s_n and s_{n+1} to approximate

Figure 2.6. Local features detected by the Hessian-Laplace detector on images with scale change

Figure 2.7. Local features detected by the Hessian-Affine detector from two different view

the Laplacian in scale space:

$$D(\mathbf{x}, s_n) = (G(\mathbf{x}, s_n) - G(\mathbf{x}, s_{n+1})) \otimes I(\mathbf{x}) \quad (2.5)$$

$$G(\mathbf{x}, s_n) = G((x, y), s_n) = \frac{1}{2\pi s^2} e^{-(x^2+y^2)/2s^2} \quad (2.6)$$

In (2.5) G is the variable-scaled Gaussian of scale s (see also (2.6)), I is the image intensity at $\mathbf{x} = (x, y)$, and $\otimes$ denotes the convolution operation. The principle of the DoG for scale selection is nearly the same as for the Harris-Laplace detector, which includes an accurate key point localization procedure, elimination of edge responses by a Hessian-matrix based analysis and orientation assignment with orientation histograms. Thus the DoG detector shows similar behavior like

Figure 2.8. Local features detected by the DoG detector.

the Hessian matrix-based detector and detects blob-like structures. The main advantage of the DoG detector is efficient and the scale invariance property, as shown in Fig. 2.8.

4) *Maximally Stable Extremal Regions* detector (MSER)

Maximally Stable Extremal Regions (MSER) detector [29] is a watershed-like algorithm based on intensity value-connected component analysis of an appropriately thresholded image. The obtained regions are of arbitrary shape and they are defined as the regions where all the intensity values within the region are consistently lower or higher with respect to the surrounding. Take a simple instance: consider all possible binary thresholds of a gray-level image. All the pixels with an intensity below the threshold are set to 0 (black), while all the other pixels are set to 1 (white). If we imagine a movie showing all the binary images with increasing thresholds, we would initially see a totally white image. As the threshold gets higher, black pixels and regions corresponding to local intensity minima will appear and grow continuously. Sometimes certain regions do not change their shape even for set of different consecutive thresholds. These are the *Maximally Stable Extremal Regions* detected by the algorithm. Finally, the irregularly shaped region detected by MSER should be replaced by an ellipse having the same shape moments to achieve affine invariant, as shown in Fig. 2.9. MSER works best for structured images which

Figure 2.9. Local features detected by the MSER detector.

can be segmented well and it is also currently the most efficient among the affine invariant feature detectors.

All feature detectors and other detectors mentioned above which are not reviewed in this thesis have been used widely in different application domain, which have also been evaluated and compared in references [25, 34, 36]. Generally, such blob detectors, corner detectors or region detectors with different invariance properties are often used together in order to have the image content better represented and thus achieve to a superior performance.

2.1.2 Feature Description

Once the local features in image such as interest points or local image patches are detected, feature description which serves as the second step in the BoF model will be applied to depict the local neighborhood region of the salient points or patches as high-dimensional descriptor vectors. Such descriptors should be distinctive, and also keep invariant and robust to various image transformations like affine distortions, scale changes, illumination changes or occlusions. Common feature descriptors include *differential invariants* [37], *steerable filters* [38], *moment invariants* [39], *complex filters* [40], *shape context* [41], *spin images* [42], the recently developed

Scale Invariant Feature Transform (SIFT) [1] and its extensions such as PCA-SIFT [43], and *gradient location and orientation histogram* (GLOH) [33]. According to a fairly comprehensive review on these various feature descriptors [33], SIFT-based descriptors (SIFT, PCA-SIFT, GLOH) outperform all the other descriptors as they are shown to be robust to various image transformations like rotation, scale changes, affine transformations and illumination changes. The robustness and the distinctive characteristics of the SIFT-based descriptors make them a good choice to be used as the feature descriptors in the BoF model. The detailed mathematical investigation of SIFT-based descriptors is provided as follows.

Developed by David Lowe [1], SIFT descriptor, which belongs to a kind of distribution-based methods that represent certain region properties by multidimensional histograms, is a 3D histogram of gradient location and orientation, where location is quantized into a 4×4 location grid and the gradient angle is quantized into eight orientations. Thus, the resulting descriptor is of dimension $4 \times 4 \times 8=128$. For details, there are two steps for computing SIFT descriptor: Orientation assignment for interested points and 128 dimensional descriptor vector generation. In the first step, the gradient magnitudes m and local orientations ϕ at the interest point $I(x, y)$ should be firstly calculated as,

$$m = \sqrt{(I(x+1, y) - I(x-1, y))^2 + (I(x, y+1) - I(x, y-1))^2} \quad (2.7)$$

$$\phi = \tan^{-1}((I(x, y+1) + I(x, y-1))/(I(x+1, y) - I(x-1, y))) \quad (2.8)$$

Both the magnitude and orientation calculations for the gradient are from the Gaussian-smoothed image $L(x, y, \sigma)$ at the point's estimated scale σ so that all computations are performed in a scale-invariant manner. Then an orientation histogram with 36 bins is formed by adding each sample in a neighboring Gaussian-weighted circular window, with each bin covering 10 degrees. The peaks in this histogram correspond to dominant orientations. Once the histogram is filled, the orientations

Figure 2.10. Illustration of the SIFT descriptor partially taken from [1] and [2]. After orientation assignment for interested points, a histogram of gradient orientations is formed in each grid cell. 2×2 grid is depicted for a better facility of illustration. 8 orientation bins are used in each grid cell therefore giving a descriptor of dimension $32 = (2 \times 2 \times 8)$.

corresponding to the highest peak and local peaks that are within 80% of the highest peaks are assigned to the interest point. In the case of multiple orientations being assigned, an additional interest point is created having the same location and scale as the original interest point for each additional orientation. This first step is to assign orientations to points at particular scales, which ensured invariance to image location, scale and rotation. After that, in the second step, the descriptor vectors is computed as a set of orientation histograms on (4×4) pixel neighborhoods. These histograms contain 8 bins each, and each descriptor contains a 4×4 array of 16 histograms around the interest point. This leads to a SIFT feature vector with $4 \times 4 \times 8 = 128$ elements. This vector is finally normalized to enhance invariance to changes in illumination. The computation process of SIFT descriptor is illustrated in Fig. 2.10.

Motivated by the problem that the dimension of the SIFT descriptor (i.e., 128

dimension) seems high, Yan and Sukthankar [43] modified the SIFT descriptor by reducing the dimensionality with *Principal Component Analysis* (PCA). Instead of gradient histograms on detected interested points, the authors applied PCA [44] to the scale-normalized gradient patches obtained by the feature detector. In principle they extract a 41×41 patch at the given scale centered on an interested point, but instead of a histogram, they describe the patch of local gradient orientations with a PCA representation of the most significant eigenvectors, that is, the eigenvectors corresponding to the highest eigenvalues. In practice, it was shown that the first 20 eigenvectors are sufficient for a proper representation of the patch. In contrast to SIFT, although it performs slightly worse in some cases, PCA-SIFT descriptor always lead to more efficient work than the standard SIFT representation [43].

Another extension of SIFT descriptor is *gradient location-orientation histograms* (GLOH), which can also obtain higher robustness and distinctiveness properties. Instead of dividing the patch around the interested points into a 4×4 regular grid, Mikolajczyk and Schmid [33] divided the patch into a radial and angular grid, which is similar to the idea that used for *shape context*. In particular, see the Fig. 2.11, the patch is divided into 3 radial and 8 angular sub-patches leading to 17 location patches. Gradient orientations of those patches are quantized to 16 bin histograms, which in fact results in a 272 dimensional descriptor. This high dimensional descriptor is reduced by applying PCA and the 128 eigenvectors corresponding to the 128 largest eigenvalues are taken for description.

There has been an extensive study [33] done on the performance evaluation of different local descriptors, including SIFT and its extensions, using a range of feature detectors. These evaluations suggest strongly that SIFT-based descriptors, which are the most robust and distinctive, outperform other local descriptors on numerous applications, and are therefore best suited for utilizing in the BoF model.

Figure 2.11. Illustration of the GLOH descriptor taken from [2].

2.1.3 Clustering and Visual Vocabulary Formulation

The third step, clustering the feature descriptors in their feature space into a large number of clusters then formulating a visual vocabulary is considered as the most important step in the BoF as this step has been shown that it can have a significant impact on the final performance of the BoF. Different from the way that the vocabulary is obtained in the case of text, which naturally results from the words occurring in the document collections, in image case, the construction of the visual vocabulary involves several variables, such as how to choose a suitable and efficient clustering algorithms especially for a large quantity of descriptors, and the problem of setting an optimal vocabulary size. In view of this, many studies about these issues will be discussed in this section.

For the first issue, it is well known that generating clusters for a large quantity of descriptor vectors presents challenges to traditionally used algorithms [18]. The size of the data essentially rules out methods such as *mean-shift* [45] and *spectral clustering* [46]. *k*-means [15] is the most popular clustering algorithm used in both the standard BoF model and its extensions due to its simplicity for implementation [7, 11, 21]. However, the traditional “flat” *k*-means clustering also has difficulty to scale to large vocabularies which results in the high computational complexity. Nister and Stewenius [3] further developed “flat” *k*-means to propose a hierarchical *k*-means algorithm based on a vocabulary tree and thus a much larger and

Figure 2.12. Illustration of the process of building the vocabulary tree taken from [3]. The hierarchical quantization is defined at each level by k clusters (in this case $k = 3$).

more discriminative vocabulary can be used efficiently which can improve the image searching quality dramatically. In hierarchical k -means, instead of k defining the final number of clusters, k defines the branch factor (number of children of each node) of the tree. First, an initial k -means process is run on the training data, defining k cluster centers. The training data is then partitioned into k groups, where each group consists of the descriptor vectors closest to a particular cluster center. The same process is then recursively applied to each group of descriptor vectors, recursively defining clusters by splitting each cluster into k new parts. The tree is determined level by level, up to some maximum number of levels L , and results in k^L clusters. Fig. 2.12 shows the illustration of the hierarchical k -means algorithm.

Similar the work of hierarchical k -means which can reduce the computational complexity to constructing large scale vocabulary, Philbin *et al.* [18] also exploited an approximate k -means clustering which has similar complexity to the vocabulary tree used in hierarchical k -means, but far superior performance. They noticed that in typical “flat” k -means, the vast majority of computation time is spent on calculating nearest neighbors between the descriptors and cluster centers. Thus they replace this exact computation by an approximate nearest neighbor method, and use a forest of 8 randomized k - d trees [47] built over the cluster centers at the beginning of each iteration to increase speed.

Some other work has been done to solve the problem of constructing a suitable

visual vocabulary to improve the performance. Moosmann *et al.* [48] employed a forest of random trees to rapidly and distinctively assign descriptors to clusters. Schindler *et al.* [49] used the same data structure as the vocabulary tree for large-scale location recognition application. In particular, they presented a *Greedy N-Best Paths* (GNP) algorithm to improve the retrieval performance of the traditional vocabulary tree algorithm by considering more candidates instead of one at each level of the tree. Chum *et al.* [50] brought the query expansion technology, which is a standard method in text retrieval system, into the visual domain for improving the retrieval performance. They utilized the spatial constraint between query image and each returned image and used these verified images to learn a latent feature model which controlled the construction of expanded queries.

Secondly, unlike the vocabulary of a text corpus whose size is relatively fixed, in image case, since the vocabulary is built using a clustering algorithm, another issue should be addressed is how to tune the vocabulary size, that is, to set a specific number of cluster so that it can result in a superior performance. Generally, clustering algorithms used in the BoF model can not determine automatically the optimal number of clusters because of the problem of large quantity of descriptor vectors we mentioned before. So setting a suitable size of vocabulary is critical since it can change the quality of the vocabulary, and also the performance of the BoF model. Choosing the right vocabulary size involves the trade-off between discriminativity and generalizability. With a small vocabulary, the visual word is not very discriminative because dissimilar descriptors can map to the same visual word. As the vocabulary size increases, the visual word becomes more discriminative, but meanwhile less generalizable and forgiving to noises, since similar descriptors maybe map to different visual words. However, although some recent studies tend to show that larger vocabulary improve the quality of the BoF model [3, 18], using a large vocabulary also increases the cost associated with clustering descriptors, assigning visual

words, and running supervised classifiers. The vocabulary size used in most work varies from several hundreds, to thousands and tens of thousands. For instance, Lazebnik *et al.* [22] adopted 200-400 visual words, Zhang *et al.* [51] adopted 1000, Sivic *et al.* [11] adopted 6,000-10,000, etc. Their results are not directly comparable due to the difference on corpus and classification methods.

Although there is still no consensus as to setting the appropriate size of a visual vocabulary, sometimes this issue can be sidestepped. For instance, Jiang *et al.* [52] made an interesting observation that once employing the suitable weighting schemes on the BoF, the impact of vocabulary size turns to be insignificant. So it is believed that more studies from all kinds of aspects on this issue will be discussed in the future.

2.1.4 Visual Words Assignment

The final step for the BoF model is to assign visual word selected from the visual vocabulary to each descriptor then representing the image by frequency counts of the visual word indices contained in them. For such visual word assignment, a *nearest neighbor* quantization process is the most commonly used way by representing each descriptor vector as the index of its nearest visual word vector [7,11,18,22]. Based on such simple nearest visual word findings, two image features are considered identical if they are assigned to the same visual word (cluster center). On the other hand, two features assigned to different (even very close) clusters are considered totally different. In effect the *nearest neighbor* quantization cannot achieve an “ideal” discriminative quantizer as it provides a very coarse approximation to the actual distance between the two features-zero if assigned to the same visual word, and infinite otherwise, and also fails to retrieve the discriminate information contained in the feature vectors which belong to the same visual word. Therefore, in recent

literature, the problem of discriminatively design of this visual word assignment process has been gaining increasing attention [4, 5].

Philbin *et al.* [4] introduced the soft-assignment technology so that a high dimensional descriptor can be mapped to a weighted combination of visual words, rather than hard-assigned to a single word. They describe *descriptor-space soft-assignment* which extract a single descriptor from each image patch and assign it to several visual words nearby in the descriptor space to achieve this weighted combination. Thus, they Changes the representation of a visual feature vector from a single word to a r -vector. Fig. 2.13 shows the benefits of soft-assignment technology [4]. Points A-E represent cluster centers (visual words), and points 1-4 are features. This figure demonstrates two benefits of soft-assignment: (1) In hard-assignment, features 3 and 4 will never be matched as they are assigned to different visual words despite being close in feature space. Using soft-assignment, words 3 and 4 will be assigned to A, B and C (with certain weights) and can be matched strongly as they are close in the feature space; (2) In hard-assignment, features 1-3 are all assigned to word A equally and there is no way of distinguishing that 2 and 3 are closer than 1 and 3. Soft-assignment provides a way of recording this information, and subsequently giving more weight to the closer matches and less to the farther. This method improved the performance of retrieval, since that it allowed the inclusion of features which were lost in quantization step based on the hard-assigned methods.

By incorporating the techniques from the fields of supervised quantizer design and information-theoretic clustering, Lazebnik [5, 6] proposed a technique for jointly quantizing continuous features and the posterior distributions of their visual words based on minimizing empirical information loss. Fig. 2.14 gives the illustration of their work. X is a continuous random variable representing the features, K is a discrete random variable representing the index of the visual words nearest to

Figure 2.13. The benefits of soft-assignment technology [4]

Figure 2.14. Illustration of the work of [5, 6].

X with respect to the visual vocabulary in the feature space, and Y is a discrete random variable representing the attribute of X that we want to predict. In an ideal case, K would be a sufficient statistic of X for Y , i.e., we would be able to predict Y based on K as well as we could have from X itself. However, some information is bound to be lost in going from the continuous feature space to the finite space of quantizer indices, and the goal of [5] is to learn a visual vocabulary that minimizes the loss of information about Y incurred by this operation.

A good quantization process for assigning visual words to continuous features should fulfill the following objective that the quantized representation should retain as much distinctive information as possible. Although some efforts have been devoted on this issue, however, compared with other study such as feature detection, summarizing image regions using invariant descriptors, there is still very little literature discussing this issue.

2.1.5 Discussion

The BoF image representation has been widely tested in image related tasks such as natural scene categorization, object recognition (instance and category level) and so on [7, 11, 20–22]. However, there are still some shortcomings with most of the existing approaches for BoF. On the one hand, BoF ignores the spatial relationships among the features, which is very important in image representation. Researchers have proposed several methods to incorporate the spatial information. For feature level improvements, correlogram features can capture spatial co-occurrences of features [21]. For generative models on learning and recognition algorithm level improvement, relative positions [53] of visual words are also taken into account. The hierarchical shape and appearance model for human action [54] introduces a new part layer (Constellation model) between the mixture proportion and the BoF features, which captures the spatial relationships among parts in the layer. For discriminative models, pyramid match kernel [20] is a fast kernel function which maps the BoF features to multi-resolution histograms. One of the advantages of the multi-resolution histograms is the ability to capture the co-occurring features. The pyramid match kernel builds the multi-resolution histogram by binning data points into discrete regions of increasing larger size. Thus, points do not match at high resolutions have the chance to match at low resolutions. Pyramid match kernel performs approximate similarity match, without explicit search, and the computation time is only linear in the number of features. Compared with other kernel approaches, pyramid match kernel is much faster, yet provides competitively accurate results. And based on pyramid match kernel, spatial pyramid match [22] is proposed to perform pyramid matching by partitioning the image into increasingly fine sub-regions and compute histograms of local features inside each sub-region, which also give promising results.

On the other hand, several problems with the current BoF exist in the unsupervised construction of the visual vocabulary and the visual word assignment process, which are unable to improve the discriminativity of the BoF. Recently, numerous approaches have been proposed to address these problems through different ways. Farquhar *et al.* [55] construct *class-specific* visual vocabularies using the *Maximum A Posterior* (MAP) approach. Winn *et al.* [56] reduce a large vocabulary constructed by a clustering algorithm using pair-wise word merging. Larlus and Jurie [57] propose a generative model that integrates visual vocabulary construction with classifier training. Perronnin *et al.* [58] characterize images using a set of category-specific histograms, where each histogram describes whether the content can best be modeled by the universal vocabulary or by its corresponding category vocabulary. In particular, because the BoF share some properties with the BoW model commonly used in text retrieval, text-based weighting methods can therefore be used to perform and improve the BoF image representation. For instance, *term frequency-inverse document frequency* (TF-IDF) weighting have been used in many cases of image tasks [11, 18]. However, due to the fundamental differences existed between text words and visual words in image (one text word usually corresponds to one given concept, whereas one visual word corresponds to a part of an object, or potentially to parts of several different objects), TF-IDF weighting method just considers that a word occurs independently from the others, which does not seem very realistic in the image cases.

Motivated by this, our work in this thesis will exploit a new weighting scheme which incorporate the contextual information among visual words into the conventional BoF model. In our BoF model, Harris-Affine detectors and SIFT descriptors, which has been widely used nowadays, are used in feature detection and feature description, respectively. The *k*-means clustering is performed to construct a visual vocabulary. After that, a new weighting scheme based on *FeatureRank* (FR)-a novel

feature weighting method which draws the idea from Brin and Page's PageRank algorithm are conducted. It is believed that this FeatureRank-based weighting scheme for the BoF model incorporating the contextual information among visual words into the conventional BoF model may lead to superior recognition results. We will introduce in the following section an brief overview of Brin and Page's PageRank algorithm and the next chapter we will provide more details on the novelly proposed FeatureRank and the FeatureRank-based weighting scheme for the BoF.

2.2 Brin and Page's PageRank Algorithm

By exploiting the graph theory to the huge web of hyperlinks, *PageRank* (PR) algorithm was developed by Brin and Page in 1998 [19], which quantitatively reflects the relative importance of web pages based on the following thesis: A web page is considered '*important*' if it is pointed (or *recommended*) by other '*important*' web pages. Here, the "recommendation" is realized by the hyperlink relationship between web pages. That is, a hyperlink pointed from the web page X to the web page Y is a recommendation from X to Y ; in this example, the hyperlink is viewed as an *inlink* of page Y , but an *outlink* of page X . Furthermore, each recommendation should be weighted by both the status of the recommender and total number of recommendations made by this recommender. In other words, a recommendation from X to Y should be weighted by both the importance value of X and the total number of outlinks from X . Thus, the importance score reflected by the *PageRank* (PR) value of web page Y is the sum of such weighted recommendations from all its inlinks. Thus, the PR value (denoted as $pr(P_i)$) of a page P_i is defined as [19]:

$$pr(P_i) = \sum_{P_j \in I_{P_i}} \frac{pr(P_j)}{|P_j|} \quad (2.9)$$

where I_{P_i} is the set of pages pointing to P_i (i.e., the inlinks of P_i), P_j is one of the inlinks of P_i , and $|P_j|$ is the total number of outlinks from page P_j . However, the $pr(P_j)$ values in (2.9) is unknown. To solve this problem, an iterative procedure is applied to (2.9) with an assumption that all pages have equal PR value in the beginning; that is, $pr_0(P_i) = 1/n$, where n is the total number of web pages in the web. (2.9) is then successively computed by substituting the values of $pr(P_j)$ obtained from the previous iteration. That is

$$pr_{k+1}(P_i) = \sum_{P_j \in I_{P_i}} \frac{pr_k(P_j)}{|P_j|} \quad (2.10)$$

where $pr_{k+1}(P_i)$ is the PR value of page P_i computed at iteration $k + 1$. However, note that (2.10) can only compute PR value for one page at a time, so in order to simultaneously calculate PR values for all web pages, a matrix representation by introducing a $n \times n$ matrix $\mathbf{H}$ and a $1 \times n$ vector $\mathbf{pr}$ is exploited, where the matrix $\mathbf{H}$ is a row normalized hyperlink matrix with $H_{ji} = 1/|P_j|$ if there is a hyperlink from page P_j to page P_i , and 0, otherwise. And the $\mathbf{pr}$ vector records the PR value of all web pages at each iteration. Thus, the iterative process presented by (2.10) can be written as:

$$\mathbf{pr}_{k+1}^T = \mathbf{pr}_k^T \mathbf{H} \quad (2.11)$$

where the $\mathbf{pr}_{k+1}^T$ is the transpose of PR value vector of all web pages at iteration $k+1$. Equation (2.11) can be modeled by computing the stationary vector $\mathbf{pr}$ of a Markov Chain with transition probability matrix $\mathbf{H}$ [59, 60]. It is known that in Markov Chain theory, for any starting vector, a numerical solution called Power Method applied to a Markov transition probability matrix will converge to a unique positive stationary vector as long as the matrix is stochastic, irreducible and aperiodic [61]. Therefore, in order to guarantee the convergence of PR value computing, $\mathbf{H}$ should be slightly adjusted for satisfying these properties as follow:

$$\mathbf{G} = \alpha \mathbf{H} + (\alpha \mathbf{a} + (1 - \alpha) \mathbf{e}) 1/n \mathbf{e}^T \quad (2.12)$$

where $\mathbf{e}^T$ is the row vector of all 1s, α is a scalar between 0 and 1, $\mathbf{a}$ is a vector that $a_i = 1$ if the page P_i with no outlinks. After these adjustments, the (2.11) can be converted to:

$$\mathbf{pr}_{k+1}^T = \mathbf{pr}_k^T \mathbf{G} \quad (2.13)$$

On the basis of (2.13), the PageRank value vector can be computed by iteratively applying the Power Method to $\mathbf{G}$. The detailed derivation steps of (2.12) and more mathematical interpretations on Brin and Page's PageRank algorithm are documented in reference [59, 60].

Chapter 3

FeatureRank Weighting and Its Use for the BoF Model

This chapter consists of two parts to provide a comprehensive introduction of the contributions in this thesis. *FeatureRank* (FR) as a novel method for weighting features is firstly proposed with a detailed mathematic description and the proposed nonparametric FeatureRank-based clustering algorithm is also explored in first part. Furthermore, the proposed FeatureRank-based weighting scheme for the BoF model is investigated and finally, this chapter is completed by some related experimental results on these FeatureRank-based applications.

3.1 Proposed FeatureRank Weighting and Its Applications

3.1.1 FeatureRank

Feature weighting, the process of which is to assign a range of weights to discriminate features in the feature space, has been actively researched in recent years,

as it is an essential step often encountered in computer vision and machine learning for improving application’s performance [4, 62–65]. Shao *et al.* [64] proposed a *local-density-ratio* as the weight vector by computing the various neighborhood densities of features to indicate the relative dense coefficient between feature and its neighbors. Besides, Marszaek *et al.* [62] focus on weighting features by incorporating additional spatial information among the features, which can capture the contextual information among features. However, among the methods mentioned above, most existing feature weighting methods suffer from the problem of only considering the local information for weighting without incorporating the global structural relationship among features [4, 63, 65]. Thus, it is believed that weighting feature based on the combination of both local and global information will lead to a superior performance.

Inspired by the central idea of Brin and Page’s PageRank, FeatureRank, an analogue used in feature space for weighting based on a core thesis: *a feature is considered ‘important’ if it is similar to other features that have been identified ‘important’ already* is derived in this thesis. FeatureRank complies with that a feature’s importance within the feature set is both affected by the amount of similar features belong to it and their self importance. In our proposed algorithm, the former factor is achieved by constructing a hyperlink relationship for features based on similarity metric to preserve the local contextual information between features. And the latter factor is reached through learning the Brin and Page’s model to apply Markov Chain model to an iterative computing so as to get the global structural information of features in the entire feature set. So FeatureRank works for giving the feature a reasonable weight based on both the local and global contextual information between features. A detailed description on FeatureRank computation is as follow:

Given a set of features, the first step in FeatureRank computation is to construct a hyperlink relationship between features by conducting a *Nearest Neighbor Search* (NNS) algorithm [66, 67], which is for finding closest features within the set by computing the Euclidean distance between features. We adopted the Euclidean distance metric here as it is usually chosen as the similarity measurement yardstick in the NNS algorithm, and it is simple and effective as evidenced by extensive experimental results [66, 67]. Besides, Euclidean distance metric provides the most straightforward way of computing distances between features, and its faster calculation speed can also enhance the computation efficiency of FeatureRank computation. After distance computing, k nearest neighbors are then selected for each feature. Therefore, a decision function to decide whether there is a hyperlink from feature f_i to feature f_j is exploited as:

$$H_\lambda(f_i, f_j) = \begin{cases} 1 & f_j \in \lambda_i \\ 0 & \text{otherwise} \end{cases} \quad (3.1)$$

where λ_i is the k nearest samples set of f_i . It should be noticed that $H_\lambda(f_i, f_j)$ is not always equal to $H_\lambda(f_j, f_i)$, because f_i may not be in the k nearest samples set of f_j . This observation implies that the proposed hyperlink construction heavily relies on the local contextual information of features. Secondly, based on this hyperlink relationship and the FeatureRank's core thesis, the FeatureRank value of the feature f_i is defined as the sum of the FeatureRank values of all features similar with f_i , which can be mathematically expressed as:

$$fr(f_i) = \frac{\sum_{j=1}^N fr(f_j) H_\lambda(f_i, f_j)}{\sum_{j=1}^N H_\lambda(f_i, f_j)} \quad (3.2)$$

where $fr(f_i)$ is the FeatureRank value of the feature f_i , N is the total number of features in the set. After that, the same as Brin and Page's solution, an iterative

process is applied for overcoming the problem of unknown $fr(f_j)$. Thus, (3.2) is changed to an iterative presentation as:

$$fr_{k+1}(f_i) = \frac{\sum_{j=1}^N fr_k(f_j) H_{\lambda}(f_i, f_j)}{\sum_{j=1}^N H_{\lambda}(f_i, f_j)} \quad (3.3)$$

This process is initiated with $fr_0(f_i) = 1/N$, which means an equal FeatureRank value for each feature at the beginning. Inspired by PageRank's mathematic solution which was introduced in Chapter 2, it is noticed that this two equations (3.2) and (3.3) can learn from the model of PR value computing, that is, (3.3) should be firstly converted to the matrix presentation:

$$\mathbf{fr}_{k+1}^T = \mathbf{fr}_k^T \mathbf{H}_{\lambda} \quad (3.4)$$

where $\mathbf{H}_{\lambda}$ is a row normalized hyperlink matrix whose entries are given by $H_{\lambda ij} = 1$, if satisfying $H_{\lambda}(f_i, f_j) = 1$; otherwise, $H_{\lambda ij} = 0$. And the $\mathbf{fr}_{k+1}^T$ is the transpose of the FeatureRank value vector of all features at iteration $k + 1$. However, encountering the same problem in PageRank's $\mathbf{H}$ matrix mentioned in Chapter 2, $\mathbf{H}_{\lambda}$ also need an adjustment for numerical computation and convergence issue, as follows:

$$\mathbf{F}_{\lambda} = \alpha \mathbf{H}_{\lambda} + 1/n(1 - \alpha) \mathbf{e} \mathbf{e}^T \quad (3.5)$$

where $\mathbf{e}^T$ is the row vector of all 1s, α is a scalar between 0 and 1. Such adjustment is to guarantee the convergence and the unique positive stationary vector exist. And because normally $1/n$ is closer to 1, $\mathbf{F}_{\lambda}$ will hold the hyperlink structure within $\mathbf{H}_{\lambda}$. Finally, the FeatureRank vector is the stationary vector computed by iteratively using power method on a Markov Chain model. This iteration process interpreted by $\mathbf{fr}_{k+1}^T = \mathbf{fr}_k^T \mathbf{F}_{\lambda}$ captures the global structural information embodied in the $\mathbf{F}_{\lambda}$.

As the FeatureRank computation complies with the core thesis: *A feature is considered important if it is similar to other important features*, it can assign the larger

weight to the features that are more similar to most of features so as to naturally find the dense regions in the feature space. In order to visually demonstrate the performance of FeatureRank for weighting features, the illustration examples tested on two dimensional synthetic dataset (generated by MATLAB) is shown in Fig. 3.1 and Fig. 3.2. Fig. 3.1 provide two examples with regular-shaped two dimensional synthetic data such as line cluster, compact spherical and ellipsoidal clusters and its FeatureRank results. As we expected, the features in dense regions can be detected as they obtained the larger FeatureRank value, whereas the features in the regions with lower density will be assigned smaller FeatureRank value. Likewise, the simulation results of Fig. 3.2 verified that, for the symmetrical structured data which consists of ring-shaped cluster, ellipsoidal shell-shaped cluster, etc., FeatureRank also holds its behavior well, which can discover dense region features of arbitrary size, shape and location.

It should be observed that the choice of parameter k may affect the performance of FeatureRank. Therefore, we conduct a experiment to verify the effect of k . The simulation results in Fig. 3.3 indicate that for a relatively broad range of values of k (when $k=40, 70, 100$), FeatureRank can produce robust results, although in extreme cases, when k reaches too high or too low (in the simulation, when $k=200$ or $k=10$), FeatureRank could not help to separate the features. In our work, an appropriate k is derived and optimized based on numerical experiments.

3.1 Proposed FeatureRank Weighting and Its Applications

Figure 3.1. Illustration examples of FeatureRank on 2D regular-shaped synthetic data. left: 2D synthetic data ; right: 2D synthetic data weighted by FeatureRank (The different colors stand for different values).

Figure 3.2. Illustration examples of FeatureRank on 2D symmetrical structured data. left: 2D synthetic data ; right: 2D synthetic data weighted by FeatureRank (The different colors stand for different values).

3.1 Proposed FeatureRank Weighting and Its Applications

Figure 3.3. Illustration examples of FeatureRank performance with different parameter k (The different colors stand for different values).

3.1.2 Nonparametric FeatureRank-based Clustering

Many clustering algorithms easily obtain incorrect results since the employed techniques often rely upon the user correctly guessing the values for the tuning parameters. For instance, k -means relies on a priori knowledge of the number of clusters. Besides, mixture models even implicitly assume the same shape for all the clusters, which are not able to handle the complexity of a real feature space [68]. Arbitrarily structured feature space can be analyzed only by nonparametric methods since these methods do not embedded assumptions. The common rationale behind such nonparametric clustering approach is that the feature space can be regarded as the empirical probability density function (p.d.f.) of the represented parameter. Dense regions in the feature space thus correspond to local maxima of the p.d.f.. Once the location of the local maxima is determined, the cluster associated with it is then obtained.

Instead of calling for a probability density function (p.d.f.) estimation process, our proposed nonparametric FeatureRank-based clustering algorithm utilizes the property of FeatureRank that assign the larger weight to the features which are more similar to most of features in the space to detect the features in dense regions. Then it carries out a *local-max shift* procedure which mimic the *mean-shift* [45] by shifting to the feature which has local maximum FeatureRank so as to find the regions in which the majority of the features inside. such *local-max shift* operation will be repeated over all features till all iterations ended at some stationary features. These stationary features are considered as the resulting clusters, and the features associated (at least approximately) with the same stationary features are considered members of the same cluster.

The illustrations of this algorithm is shown in Fig. 3.4. Ground truth data and the input data are given as Fig. 3.4 (a) and (b), respectively. Firstly, the set of

input features are weighted by FeatureRank (As the data is 2D, the distance metric adopted here is Euclidean distance), let X be the feature set in n -dimensional Euclidean space, $\mathbf{R}^n$, denote the any component of X by x , and denote the FeatureRank value calculated for each feature in X as fr_x , as shown in Fig. 3.4 (c), where different colors denote various FeatureRank values. Then a fixed-radius window is created for each feature, consider x as an example, let $S_x \subset X$ be this window with the center as x , thus, the *local-max* of FeatureRank value fr with S_x is defined as: $F_{max}(x) = \arg \max_{x_k \in S_x} (fr_{x_k})$, where x_k is components of S_x . Then a proposed *local-max shift* operation is applied by shifting from the initial start feature x to $F_{max}(x)$ which has local maximum FeatureRank so as to find features in the direction of the maximum increase within the window, as shown in Fig. 3.4 (d). This *local-max shift* operation can be mathematically expressed as, $x \rightarrow F_{max}(x)$. Then the window is migrated to a new position and iteratively shift to the local maximum FeatureRank within the current window till reached a stationary one, which is $F_{max}(x) = x$. After that, change the start point and then repeat the local-max shift over all features. Such iterations will repeated till all features has reached a stationary local maximum, as shown in Fig. 3.4 (e) and (f). Due to successive computation and shift of *local-max* of FeatureRank, for each feature x , a sequent $x, F_{max}(x), F_{max}(F_{max}(x)), \dots$, which is called the *trajectory* of x (Fig. 3.4 (g)) is defined as a path leading to a stationary local maximum. According to this, the features associated (at least approximately) with the same stationary local maximum are considered members of the same cluster. And the stationary local maximums are therefore denoted as the cluster centers (Fig. 3.4 (h) gives the final clustering results, the features marked by “+” is the cluster centers).

In summary, the nonparametric FeatureRank-based clustering relies on the following steps: 1) compute FeatureRank for all features; 2) iteratively apply the *local-max shift* procedure to all features by successive finding the local maximum of

FeatureRank value till all iterations ended to be stationary; 3) cluster the features according to their *local-max shift trajectories*.

For visualization, we apply the nonparametric FeatureRank-based clustering algorithm to different 2D synthetic datasets (generated by MATLAB, two regular-shaped 2D synthetic datasets with line cluster, compact spherical and ellipsoidal clusters, and two symmetrical structured datasets with ring-shaped clusters, ellipsoidal shell-shaped clusters) so as to demonstrate the performance of this novel proposed clustering method. Comparisons with conventional k -means clustering algorithm, kernel k -means clustering algorithm [69] on different 2D synthetic datasets are also conducted to verify that the proposed one can handle arbitrarily structured clusters without any embedded assumptions.

Fig. 3.5 shows the results on the first dataset, where one line cluster and two compact spherical clusters is used as test data. Apart from the drawback of the k -means clustering that it requires the number of clusters as input, it is clear to see that the inappropriate choice of initial cluster centers may yield poor clustering results. However, for the proposed algorithm, this problem is sidestepped. That is, FeatureRank has a good property to detect the initial cluster centers automatically without any knowledge of the number of clusters and guarantees a good starting for the reasonable clustering. The kernel k -means clustering algorithm (A Gaussian RBF kernel with $\sigma=1$) can also reach a comparable performance on this first dataset, however, in the second dataset presented in Fig. 3.6, it cannot separate the clusters (A Gaussian RBF kernel with $\sigma=10$). The results on other two symmetrical structured datasets also indicate that the *local-max shift* procedure in the proposed algorithm can help to discover clusters of arbitrary shape, see Fig. 3.7, 3.8. As shown in the clustering results, compared with kernel k -means, our proposed method is doing better on identifying clusters of arbitrary shape, such as ring-shaped clusters, crossed spherical clusters.

3.1 Proposed FeatureRank Weighting and Its Applications

Figure 3.4. Illustration of proposed nonparametric FeatureRank-based Clustering

3.1 Proposed FeatureRank Weighting and Its Applications

Figure 3.5. Clustering algorithms comparisons on 2D regular-shaped synthetic dataset (one line cluster and two compact spherical clusters). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=1$) (d) result of proposed nonparametric FeatureRank-based clustering

3.1 Proposed FeatureRank Weighting and Its Applications

Figure 3.6. Clustering algorithms comparisons on 2D regular-shaped synthetic dataset (two line cluster and one compact spherical clusters). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=10$) (d) result of proposed nonparametric FeatureRank-based clustering

Figure 3.7. Clustering algorithms comparisons on symmetrical structured datasets (ring-shaped cluster). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=1$) (d) result of proposed nonparametric FeatureRank-based clustering

3.1 Proposed FeatureRank Weighting and Its Applications

Figure 3.8. Clustering algorithms comparisons on symmetrical structured datasets (ellipsoidal shell-shaped cluster). (a) ground truth data (b) result of k -means (c) result of kernel k -means (A Gaussian RBF kernel with $\sigma=0.05$) (d) result of proposed nonparametric FeatureRank-based clustering

3.2 FeatureRank-based Weighting Scheme for the BoF Model

3.2.1 Detailed Weighting Scheme

The analogy between visual words in BoF model and text words in BoW model opens up opportunities for migrating text-based retrieval techniques to solve problems in image data [4, 16, 17], therefore, some weighting scheme based on text retrieval methods are employed to improve the retrieval performance in image domain, as shown in Table 3.1. In particular, the classical *term frequency-inverse document frequency* (TF-IDF) weighting method is the most popular weighting scheme used to perform visual word-based image retrieval [11, 18]. In the framework of TF-IDF, the weight value represents the word importance in the description of the document with the following assumption: the greater the weight w_{ij} is, the better the word t_i describes the document d_j . As shown in Table 3.1, in TF-IDF, The weight w_{ij} of word t_i in document d_j can be divided into two parts: *term frequency* (TF) and *inverse document frequency* (IDF). On the one hand, TF is the number of occurrences of word t_i in document d_j , which reflects the word importance inside the document and aims at emphasizing words of high frequency; on the other hand, IDF reflects the word importance in the collection, which emphasizes the words that occur in a small number of documents. The idea of IDF assumes that, the more documents a word appears in, the less discriminative this word is. On the contrary, if a word occurs in a very few documents, then it might be a good descriptor of these documents' content. Thus, TF-IDF further degrades the importance of a word in a document if the word also frequently appears in other document.

However, due to the fundamental differences existed between text words and visual words in image, the foregoing mentioned weighting schemes which migrate

Table 3.1

TEXT RETRIEVAL-BASED WEIGHTING SCHEMES FOR THE BOF MODEL (t_i : THE i TH WORD; d_j : THE j TH DOCUMENT; w_{ij} : WEIGHT FOR t_i IN DOCUMENT d_j ; tf_{ij} : THE NUMBER OF OCCURRENCES OF WORD t_i IN DOCUMENT d_j ; N : TOTAL NUMBER OF DOCUMENTS; n_i : THE NUMBER OF DOCUMENTS HAVING WORD t_i ; TF: TERM FREQUENCY; IDF: INVERSE DOCUMENT FREQUENCY.)

Name of weighting scheme	Factors	w_{ij}
Binary [70]	Binary	1 if t_i is present in d_j , 0 if not
TF [22]	TF	tf_{ij}
TF-IDF [11, 18]	TF, IDF	$tf_{ij} \cdot \log \frac{N}{n_i}$

the text-based method directly into the image cases are not realistic as they simplify the semantics behind the visual words and neglect inter-word relationship. For example in Fig. 3.9, five visual words derived from the visual vocabulary produce similarity scores equal to zero using text-based weighting methods. However, it is obviously that image 3 and 4 are more alike as both visual word 3 and 4 depict the visual parts of eyes. In other word, a visual-based weighting scheme should reflect such differences (i.e., compared with histogram2, histogram3 and histogram4 should get a different similarity score, not zero).

Some related works has been proposed to address the problem illustrated in Fig. 3.9. Motivated by text-based ontology such as the WordNet, Jiang *et al.* [71] construct a visual-based ontology to capture the hyponym (*is-a*) relationship of visual words for weighting. To address the problem of failing to retrieve the contextual information between visual words buried under the histogram bin-to-bin comparison, Jiang *et al.* [71] as well as Philbin *et al.* [4] proposed the soft-weighting scheme which performs one feature-to-many visual words mapping, while linguistically interpreting the relationship of multiple visual words in close proximity. This scheme still follows the convention of bin-to-bin comparison, but inter-word relationship is inherently modeled during the construction of the visual word histogram.

Figure 3.9. Text-based weighting scheme using in BoF which produces similarity scores equal to zero

Although these efforts on exploiting the visual words correlation-based weighting scheme, there is little work on combining the classical *term frequency-inverse document frequency* (TF-IDF) weighting method with visual word relationships. Such combination not only reflects the visual words' importance both in an image and in the entire database but also emphasizes the content-based correlation among visual words, which is believed that can result in a better performance. Inspired by this, a novel FeatureRank-based weighting scheme for the BoF model which incorporates inter-word relationship into the traditional text-based weighting scheme is proposed in this thesis.

In the proposed FeatureRank-based weighting scheme, two steps are conducted, inter-word relationship estimation and combination inter-word contextual information with TF-IDF. The following will describes these two steps in details.

Inter-word relationship estimation

Given a visual vocabulary, several relationships between visual words can be defined in a reasonable way, for instance, Jiang *et al.* [71] employ a visual-based ontology to capture the hyponym (*is-a*) relationship of visual words. In our approach, we focus on defining the relatedness by considering the different discriminative ability of visual words. Fig. 3.10 shows an example set of visual words. It is clear that visual words 3, 4 and 5 do not seem reliable as they depict almost the same parts of the object and are over-discriminative, in other words, these three visual words should be combined into one visual word. Such over-discriminative will lead to a high probability of error visual word partitions during vector quantization process of the BoF therefore result in a worse recognition result. Based on this, inspired by the property of FeatureRank that can detect the features in dense region in feature space, we utilize FeatureRank to find such kind of visual words in the visual vocabulary then assign them a lower importance weight. Let V be a vocabulary of n visual words: $V = (v_1, v_2, \dots, v_n)$, then for each visual word, FeatureRank value can be presented as, $fr_i, i \in n$. Because the FeatureRank value is inversely proportional to the importance weight, the weight value of visual word v_i can be defined as,

$$w_{v_i} = \log \frac{1}{fr_i} \quad (3.6)$$

After that, a normalization should be conducted to set all weights in a similar range so that they can be compared to rank the importance of visual words in the vocabulary.

Combination inter-word information with TF-IDF

The second step aims to achieve a combined weighting which not only emphasizes different contributions to the vocabulary of visual words but also reflects the visual words' importance both in the image and in the entire database. Thus, let v_i is the

Figure 3.10. The illustration of over-discriminative visual words

i th word in the vocabulary, I_j is the j th image, the resulting weight value w_{ij} is divided into two parts, so that

$$w_{ij} = w_{v_i} \cdot (tf_{ij} \cdot \log \frac{N}{n_i}) = \log \frac{1}{fr_i} \cdot (tf_{ij} \cdot \log \frac{N}{n_i}) \quad (3.7)$$

where tf_{ij} is the number of occurrences of visual word v_i in image I_j , N is the total number of images in database, and n_i is the number of images having visual word v_i . In this weight definition, tf_{ij} up-weights visual words occurring often in a particular image and the $\log \frac{N}{n_i}$ down-weights visual words that appear often in the image database, meanwhile, w_{v_i} degrades the visual words that are not reliable or over-discriminative in the visual vocabulary. Thus, each image in database can be represented by a n -vector $h_j = (w_{1j}, w_{2j}, \dots, w_{nj})$. At the retrieval stages, like the TF-IDF weighting, our scheme still follows the bin-to-bin histogram comparisons by computing the cosine similarity between query vector h_q and all images vectors h_j [72, 73].

3.2.2 Experiments

To verify the performance of our proposed weighting scheme, experiments on two database for the task of scene recognition are conducted in this section. In the following, the details will be reviewed.

Databases

4 natural scenes categories: The database 1, contains 4 categories of natural scenes (totally 400 images) which comes from the original 13 natural scenes categories database provided by [74] and [7], as shown in Fig. 3.11, where the average size of each image is approximately 250×300 pixels. Each category contains 100 images. In our experiment, 10 images for each category are randomly chosen as the training images, and the rest 90 images are all considered as queries. The images in this database are all grayscale images, thus, there is no problem of illumination variations. However, a big challenge for this database is how to get a good vocabulary containing discriminant visual words to solve the problem of intra-class variations, as some images of different categories in this database are very similar, such as the inside of cities category and street category.

Paris Dataset: The database 2, Paris Dataset, consists of 6412 images collected from *Flickr* by searching for 12 particular Paris locations, where the average size of each image is approximately 768×1024 pixels [4]. In our experiment, we select some images of the 9 location categories, as shown in Fig. 3.12, which totally consists of 1080 images and each category contains 120 images. Different from the case of the database 1, this database have little intra-class variations as some images within one location category are very similar, but they may vary in position, orientation, scale or illumination. For this database, 20 images for each category are randomly chosen as the training images, and the rest 100 images are all considered as queries.

Figure 3.11. Database1: 4 natural scenes categories from [7], (a) inside of cities, (b) streets, (c) suburb residence, (d) tall buildings

Implementation

It is known that for BoF based visual search tasks, building up an optimal visual vocabulary is an awkward issue because it depends on the database considered and on the different parameters of the vocabulary building process such as the types of feature detectors, the choice of clustering algorithms, the vocabulary size and so on. Our work in this thesis aims at verifying the performance of the proposed weighting scheme used in BoF model, thus, finding the best vocabulary possible is not absolutely necessary. We prefer a common method to build our vocabularies and evaluate the results on different vocabulary sizes instead of finding an optimal

Figure 3.12. Database2: Paris Database from [4], (a) La Defense, (b) Eiffel Tower, (c) Hotel des Invalides, (d) Louvre, (e) Moulin Rouge, (f) Arc de Triomphe, (g) Notre Dame, (h) Pantheon, (i) Sacre Coeur

one. In our experiment, we use the Harris-Affine detector and MSER detector to find the interest points and salient regions in images. And the 128-dimensional SIFT descriptor is then applied to describe these features. After that the k -means clustering is performed to construct a visual vocabulary. Then each image will be encoded as a histogram and conducted histogram bin-to-bin matching for further retrieval.

Evaluations

We evaluate the results using standard performance measures of information retrieval: *Average Precision* (AP). In the following, Rt_q denotes the total retrieval results when query image is q and Rl_q denotes the images that are relevant to q [17].

Precision:

$$P = \frac{|Rl_q \cap Rt_q|}{|Rl_q|}$$

In our experiment, we only computed the precision after the first ranked image.

Results and Discussions

To verify the performance of the proposed FeatureRank-based weighting scheme for the BoF, its comparisons with the traditional TF-IDF weighting method on two different databases are conducted. The results are summarized in Table.3.2. Fig. 3.13 and Fig. 3.14 show the average precision curve comparisons on the database of 4 natural scenes categories and Paris Dataset, respectively, which demonstrate that the proposed FeatureRank-based weighting scheme which considers the inter-word relatedness will yield superior performance to the traditional TF-IDF weighting method. Another interesting observation of 3.13 and Fig. 3.14 is that our proposed weighting method may help to reduce the impact of vocabulary size in the BoF model. With scaling up the size of the visual vocabulary, the average precision rate given by proposed FeatureRank-based weighting scheme stands in an acceptable range whereas the average precision rate given by TF-IDF weighting dropped rapidly. It is well known that there is still no consensus as to setting the appropriate size of a visual vocabulary [11,22,51], however, once employing the proposed weighting schemes on the BoF, this issue can be sidestepped. Fig. 3.15 provides the average precision comparisons of proposed FeatureRank-based weighting and TF-IDF weighting, on each location category of paris dataset(the vocabulary

Table 3.2

THE AVERAGE PRECISION COMPARISONS OF TWO WEIGHTING SCHEMES: PROPOSED FEATURERANK-BASED WEIGHTING AND TF-IDF, ON TWO DATABASES.

vocabulary size	4 natural scenes categories		vocabulary size	paris dataset	
	Proposed	TF-IDF		Proposed	TF-IDF
500	66.94%	66.3%	1000	67.1%	66.9%
600	70.56%	68.3%	1500	69.0%	67.3%
700	71.67%	70%	2000	69.6%	68.7%
800	74.72%	72.2%	2500	70.0%	69.2%
900	77.78%	75.28%	3000	70.2%	69.8%
1000	80%	78.06%	3500	71.2%	69.9%
1100	78.61%	75%	4000	71.7%	70.4%
1200	76.94%	71.11%	4500	71.5%	69.1%
1300	76.11%	69.17%	5000	71.2%	68.3%
Mean AP	74.81%	71.71%	Mean AP	70.16%	68.84%

size=4000). Although better to be tested on more location categories to verify the stability and feasibility, it is still apparent to see that proposed FeatureRank-based weighting can improve the performance for 8 out of the 9 location categories, while the performance of the other one only slightly drops. In summary, we have shown that the proposed FeatureRank-based weighting is relatively superior than the previous approach. Although the main disadvantage of using small-sized database is that it may provide much smaller set of visual words without good discrimination among them due to a small number of training images available from the dataset, it still have important implications to larger sized image databases.

Figure 3.13. The average precision curve comparisons of two weighting schemes: Proposed FeatureRank-based weighting and TF-IDF, on database of 4 natural scenes categories

Figure 3.14. The average precision curve comparisons of two weighting schemes: Proposed FeatureRank-based weighting and TF-IDF, on 9 location categories of paris dataset

Figure 3.15. The average precision comparisons of two weighting schemes: Proposed FeatureRank-based weighting and TF-IDF, on each location category of paris dataset (the vocabulary size=4000)

Chapter 4

Conclusion and Future Work

4.1 Conclusion

In this thesis, a new feature weighting method called the *FeatureRank* (FR), was firstly exploited. Secondly, a nonparametric clustering based on FeatureRank was then developed to verify the feasibility of FR. Finally, a BoF model with proposed FeatureRank-based weighting scheme was built up. The conclusions are summarized below:

Firstly, *FeatureRank* (FR), a novel method for weighting in the feature space, is mathematically derived based on a core thesis analogous to ranking web pages by popularity in Google's *PageRank* (PR), wherein *a feature is considered important if it is similar to the other important features*.

Furthermore, by means of FeatureRank assigning the larger weight to the features in dense regions of the feature space, a nonparametric FeatureRank-based clustering algorithm without calling for a priori knowledge of the number of clusters are developed. Extensive demonstration examples on two dimensional datasets show that the established FeatureRank and the nonparametric FeatureRank-based clustering algorithm has achieved effective results.

Finally, FeatureRank-based weighting scheme for *bag of features* (BoF) model is investigated in this thesis. The proposed BoF model uses Harris-Affine detectors and SIFT descriptors, which has been widely used nowadays, for feature detection and feature description, respectively. The k -means clustering is then performed to construct a visual vocabulary. After that, a new weighting scheme based on proposed *FeatureRank* (FR) are conducted. Unlike the traditional BoF model, which suffers from the problem of ignoring contextual information among *visual words*, our proposed BoF model with FeatureRank-based weighting scheme can address this issue. FeatureRank-based weighting scheme use FeatureRank to distinguish each visual word so that the contextual information among visual words was included in the BoF model. Experimental results based on two different databases for scene recognition indicate that the BoF model with established FeatureRank-based weighting scheme has yielded superior performance.

4.2 Future Work

Based on the research in this thesis, the following is a suggested list of research areas to be focused in the future:

- **Combination of global and local features:** The current feature detection and description used in the BoF model, based on local features, requires at least lightly salient parts in image. However, the visual search and recognition tasks is more often done based on global image features such as color histograms and color moments, texture features based on wavelet or Gabor filters [75], etc. Local features describe the local structures in an image and do not contain color information, whereas global features are statistics about the overall distribution of color, texture, or edge information in the image.

In view of this, it is believed that a significant performance gain can be realized from further development of effective representations which combine this two types of features. Thus, the next step may be developing the combined multiple visual representations which not only considers the local structure in image but also incorporates the information of global feature distribution as a complementary. Initial work in this direction has been done by Opelt *et al.* [76] and Yang *et al.* [77].

- **Semantic visual vocabulary construction:** In most applications of the BoF model, the visual vocabulary construction relies on the simple k -means clustering approach without considering the semantic inside the visual words. Thus, another interesting research direction would be investigating language models-based visual vocabulary that can describe the distribution of visual words in images to incorporate the semantic relationship between visual words. Existing language modeling techniques for Information Retrieval (e.g., *bigram* or *trigram* type of models) can be employed in image domain to help providing a generative view of images so as to explore the semantic information among visual words as well as to capture the spatial relationships of adjacent visual words, which could be more powerful in terms of describing complex image content [78].
- **Combination of text-based retrieval method and visual properties:** The analogy between visual words in BoF model and words in BoW model opens up opportunities for migrating text-based retrieval techniques to solve problems in image data. However, current text-based approaches using in the BoF suffer from the problems of ignoring the fundamental semantic differences existing between visual words and textual words, such as the way of the vocabulary obtained, the meaning of words, the influence of the vocabulary

size and so on. Thus, for future research, a combination of textual retrieval method and visual properties which can explore more semantic information as a supplement for the BoF could therefore lead to effective and efficient visual search and recognition performance [17].

- **FeatureRank-based Applications:** The effectiveness of the established FeatureRank as a weighting method in feature space has been shown through the demonstration examples of two dimensional synthetic data. In addition, the performance of the established nonparametric FeatureRank-based clustering algorithm is verified by two dimensional synthetic datasets. One direction of future work is to explore more applications of both the FeatureRank and the nonparametric FeatureRank-based clustering algorithm, for example image segmentation. As for the established FeatureRank-based weighting scheme for the BoF model, future work might be focused on scaling-up the size and the type of the database in order to verify the robustness of this weighting scheme under various applications.

Appendix A

The Bag of Word (BoW) Model

The *bag of word* (BoW) model [14] is a popular method for representing a text document based on a dictionary-based modelling approach. In this BoW model, each document is considered as a “bag”, which contains the words from a dictionary, while the grammar and even the order of words are completely disregarded. Using the indexes of the dictionary, each bag is represented by a vector which records the count of the corresponding word in the dictionary. The following is an example of BoW representation.

In the following, three text documents are provided:

1. *Based on keypoints extracted as local image patches, an image can be described as a bag of features and this representation has been frequently used in scene classification.*
2. *Bag of Features deriving from local keypoints in image has recently appeared promising for object and scene classification.*
3. *With the widespread use of next generation mobile phones a large percentage of the population carries a potent processing unit, which nowadays is accompanied by a digital camera.*

Based on these three text documents, a dictionary can be constructed as follows:

[1: keypoints; 2: local; 3: image; 4: described; 5: bag; 6: features; 7: representation; 8: scene; 9: classification; 10: object; 11: mobile; 12: percentage; 13: phones; 14: population; 15: digital; 16: camera]

Using the indexes of the dictionary, each document is represented by a 16-entry vector as illustrated in this example. For instance, counting the word “image” in the first document is 2, so the third entry (according to the index in this dictionary, which is 3) in the 16-entry vector of the first document is 2. Thus, the BoW representations of these three documents are as follows:

1. [1,1,2,1,1,1,1,1,0,0,0,0,0,0,0]
2. [1,1,1,0,1,1,0,1,1,0,0,0,0,0,0]
3. [0,0,0,0,0,0,0,0,0,1,1,1,1,1,1]

The BoW representation compares the occurrences of each word encountered in the text, instead of comparing words by words for matching two documents. In the above example, it is clear to see that the first two documents share the similar topic due to their similar 16-entry vectors. Thus, the BoW representation method is attractive when compared with other approaches in that: it enables efficient search with simple implementation of vector matching, especially in the large text collections.

Bibliography

- [1] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [2] P. Roth and M. Winter, “Survey of appearance-based methods for object recognition,” Institute for Computer Graphics and Vision, Graz University of Technology, Austria, Tech. Rep., 2008.
- [3] D. Nister and H. Stewenius, “Scalable recognition with a vocabulary tree,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, New York, USA, June 2006, pp. 2161–2168.
- [4] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Lost in quantization: Improving particular object retrieval in large scale image databases,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, USA, June 2008, pp. 1–8.
- [5] S. Lazebnik and R. Maxim, “Supervised learning of quantizer codebooks by information loss minimization,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 7, pp. 1294–1309, 2009.
- [6] S. Lazebnik and M. Raginsky, “Learning nearest-neighbor quantizers from labeled data by information loss minimization,” in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, San Juan, Puerto

- Rico, March 2007, pp. 1–8.
- [7] F.-F. Li and P. Perona, “A Bayesian hierarchical model for learning natural scene categories,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, USA, June 2005, pp. 524–531.
- [8] J. Sivic, “Efficient visual search of images and videos,” Ph.D. dissertation, University of Oxford, 2006.
- [9] N. Davies, K. Cheverst, A. Dix, and A. Hesse, “Understanding the role of image recognition in mobile tour guides,” in *Proceedings of the International Conference on Human Computer Interaction with Mobile Devices and Services*, Salzburg, Austria, September 2005, pp. 191–198.
- [10] B. Johansson and R. Cipolla, “A system for automatic pose-estimation from a single image in a city scene,” in *Proceedings of International Conference on Signal Processing, Pattern Recognition and Applications*, Crete, Greece, June 2002, pp. 1–8.
- [11] J. Sivic and A. Zisserman, “Video Google: A text retrieval approach to object matching in videos,” in *Proceedings of the International Conference on Computer Vision*, Nice, France, October 2003, pp. 1470–1477.
- [12] D. Huttenlocher and S. Ullman, “Recognizing solid objects by alignment with an image,” *International Journal of Computer Vision*, vol. 5, no. 2, pp. 195–212, 1990.
- [13] D. Lowe., “The viewpoint consistency constraint,” *International Journal of Computer Vision*, vol. 1, no. 1, pp. 57–72, 1987.

- [14] D. Lewis, “Naive Bayes at forty: the independence assumption in information retrieval,” in *Proceedings of the European Conference on Machine Learning*, Chemnitz, Germany, April 1998, pp. 4–15.
- [15] J. Hartigan and M. Wong, “A k-means clustering algorithm,” *Applied Statistics*, vol. 28, no. 1, pp. 100–108, 1979.
- [16] J. Sivic and A. Zisserman, “Efficient visual search for objects in videos,” *Proceedings of the IEEE*, vol. 96, no. 4, pp. 548–566, 2008.
- [17] P. Tirilly, V. Claveau, and P. Gros, “A review of weighting schemes for bag of visual words image retrieval,” IRISA, France, Tech. Rep., 2009.
- [18] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, USA, June 2007, pp. 1–8.
- [19] S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” *Computer Networks and ISDN Systems*, vol. 33, pp. 107–117, April 1998.
- [20] K. Grauman and T. Darrell, “The Pyramid Match Kernel: Discriminative classification with sets of image features,” in *Proceedings of the IEEE International Conference on Computer Vision*, Beijing, China, October 2005, pp. 1458–1465.
- [21] S. Savarese, J. Winn, and A. Criminisi, “Discriminative object class models of appearance and shape by correlatons,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, New York, USA, June 2006, pp. 2033–2040.

- [22] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, New York, USA, June 2006, pp. 2169–2178.
- [23] K. Mikolajczyk and C. Schmid, “An affine invariant interest point detector,” in *Proceedings of the European Conference on Computer Vision*, Copenhagen, Denmark, May 2002, pp. 128–142.
- [24] —, “Scale and Affine invariant interest point detectors,” *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.
- [25] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schafalitzky, T. Kadir, and L. V. Gool, “A comparison of affine region detectors,” *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.
- [26] Y. Dufournaud, C. Schmid, and R. Horaud, “Matching image with different resolutions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Hilton Head, USA, June 2000, pp. 612–618.
- [27] C. Harris and M. Stephens, “A combined corner and edge detection,” in *Proceedings of the Alvey Vision Conference*, Manchester, UK, September 1988, pp. 147–151.
- [28] K. Mikolajczyk and C. Schmid, “Indexing based on scale invariant interest points,” in *Proceedings of the International Conference on Computer Vision*, Vancouver, Canada, July 2001, pp. 525–531.
- [29] J. Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide baseline stereo from maximally stable extremal regions,” in *Proceedings of the British Machine Vision Conference*, Cardiff, UK, September 2002, pp. 384–393.

- [30] T. Kadir and M. Brady, “Saliency, scale and image description,” *International Journal of Computer Vision*, vol. 45, no. 2, pp. 83–105, 2001.
- [31] T. Tuytelaars and L. V. Gool, “Wide baseline stereo matching based on local, affinely invariant regions,” in *Proceedings of the British Machine Vision Conference*, Oxford, UK, September 2000, pp. 412–425.
- [32] —, “Matching widely separated views based on affine invariant regions,” *International Journal of Computer Vision*, vol. 59, no. 1, pp. 61–85, 2004.
- [33] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.
- [34] C. Schmid, R. Mohr, and C. Bauckhage, “Evaluation of interest point detectors,” *International Journal of Computer Vision*, vol. 37, no. 2, pp. 151–172, 2000.
- [35] D. Lowe, “Object recognition from local scale-invariant features,” in *Proceedings of the IEEE International Conference on Computer Vision*, Kerkyra, Greece, September 1999, pp. 1150–1157.
- [36] T. Tuytelaars and K. Mikolajczyk, “Local invariant feature detectors: A survey,” *Foundations and Trends in Computer Graphics and Vision*, vol. 3, pp. 177–280, January 2008.
- [37] J. Koenderink and A. V. Doorn, “Representation of local geometry in the visual system,” *Biological Cybernetics*, vol. 55, pp. 367–375, March 1987.
- [38] W. Freeman and E. Adelson, “The design and use of steerable filters,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 9, pp. 891–906, 1991.

- [39] L. V. Gool, T. Moons, and D. Ungureanu, “Affine/Photometric invariants for planar intensity patterns,” in *Proceedings of the European conference on Computer Vision*, Cambridge, UK, April 1996, pp. 642–651.
- [40] F. Schaffalitzky and A. Zisserman, “Multi-view matching for unordered image sets,” in *Proceedings of the European Conference on Computer Vision*, Copenhagen, Denmark, May 2002, pp. 414–431.
- [41] S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no. 4, pp. 509–522, 2002.
- [42] S. Lazebnik, C. Schmid, and J. Ponce, “Sparse texture representation using affine-invariant neighborhoods,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Madison, USA, June 2003, pp. 319–324.
- [43] K. Yan and R. Sukthankar, “PCA-SIFT: A more distinctive representation for local image descriptors,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Washington, D.C., USA, July 2004, pp. 506–513.
- [44] I. T. Jolliffe, *Principal Component Analysis*. Springer, 2002.
- [45] D. Comaniciu and P. Meer, “Mean Shift: A robust approach toward feature space analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603 – 619, May 2002.
- [46] A. Y. Ng, M. I. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” in *Proceedings of the Advances in Neural Information Processing Systems*, Vancouver, Canada, December 2001, pp. 849–856.

- [47] V. Lepetit, P. Laguerre, and P. Fua, “Randomized trees for real-time keypoint recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, USA, June 2005, pp. 775–781.
- [48] F. Moosmann, B. Triggs, and F. Jurie, “Randomized clustering forests for building fast and discriminative visual vocabularies,” in *Proceedings of the Advances in Neural Information Processing Systems*, Vancouver, Canada, December 2006, pp. 985–992.
- [49] G. Schindler, M. Brown, and R. Szeliski, “City-scale location recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, USA, June 2007, pp. 1–7.
- [50] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, “Total Recall: Automatic query expansion with a generative feature model for object retrieval,” in *Proceedings of the International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007, pp. 1–8.
- [51] J.-G. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, “Local features and kernels for classification of texture and object categories: An In-Depth Study,” INRIA, France, Tech. Rep., 2005.
- [52] Y.-G. Jiang, C.-W. Ngo, and J. Yang, “Towards optimal bag-of-features for object categorization and semantic video retrieval,” in *Proceedings of the ACM International Conference on Image and Video Retrieval*, Amsterdam, Netherlands, July 2007, pp. 494–501.
- [53] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky, “Describing visual scenes using transformed objects and parts,” *International Journal of Computer Vision*, vol. 77, pp. 291–330, 2008.

- [54] J. Niebles and F.-F. Li, “A hierarchical model of shape and appearance for human action classification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, USA., June 2007, pp. 1–8.
- [55] J. Farquhar, S. Szedmak, H. Meng, and J. Taylor, “Improving bag-of-keypoints image categorisation,” University of Southampton, Tech. Rep., 2005.
- [56] J. Winn, A. Criminisi, and T. Minka, “Object categorization by learned universal visual dictionary,” in *Proceedings of the International Conference on Computer Vision*, Beijing, China, October 2005, pp. 1800–1807.
- [57] D. Larlus and F. Jurie, “Latent mixture vocabularies for object categorization and segmentation,” *Journal of Image and Vision Computing*, vol. 27, no. 5, pp. 523–534, 2009.
- [58] F. Perronnin, C. Dance, G. Csurka, and M. Bressian, “Adopted vocabularies for generic visual categorization,” in *Proceedings of the European Conference on Computer Vision*, Graz, Austria, May 2006, pp. 464–475.
- [59] A. N. Langville and C. D. Meyer, “Deeper inside pagerank,” *Internet Mathematics*, vol. 1, no. 3, pp. 335–400, 2004.
- [60] ———, *Google’s PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, 2006.
- [61] W. J. Stewart, *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, 1994.
- [62] M. Marszaek and C. Schmid, “Spatial weighting for bag-of-features,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, New York, USA, June 2006, pp. 2118–2125.

- [63] Y.-J. Lee and K. Grauman, “Foreground Focus: Finding meaningful features in unlabeled images,” in *Proceedings of the British Machine Vision Conference*, Leeds, UK, September 2008, pp. 1–8.
- [64] Z. Shao, X. Yang, W. Wen, and Z. Hao, “A local-density-ratio based algorithm for setting weight in weighted least squares support vector machine,” in *Proceedings of the International Conference on Intelligent Systems Design and Applications*, Jinan, China, October 2006, pp. 167–171.
- [65] D.-L. Zhao, Z.-C. Lin, and X.-O. Tang, “Contextual distance for data perception,” in *Proceedings of the International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007, pp. 1–8.
- [66] V. D. M. B. P. Zezula, G. Amato, *Similarity Search - The Metric Space Approach*. Springer, 2006.
- [67] D. G. S. Richard O. Duda, Peter E. Hart, *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.
- [68] A. Jain, R. Duin, and J.-C. Mao, “Statistical pattern recognition: A review,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.
- [69] F. Camastra and A. Verri, “A novel kernel method for clustering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 801–804, 2005.
- [70] E. Nowak, F. Jurie, and B. Triggs, “Sampling strategies for bag-of-features image classification,” in *Proceedings of the European Conference on Computer Vision*, Graz, Austria, May 2006, pp. 490–503.

- [71] Y.-G. Jiang and C.-W. Ngo, “Visual word proximity and linguistics for semantic video indexing and near-duplicate retrieval,” *Computer Vision and Image Understanding*, vol. 113, no. 3, pp. 405–414, 2009.
- [72] S. Belongie, J. Malik, and J. Puzicha, “Image retrieval using both color and texture features,” *The Journal of China Universities of Posts and Telecommunications*, vol. 14, no. 1, pp. 94–99, 2007.
- [73] S. S. A. Vadivel, A K Majumdar, “Performance comparison of distance metrics in content-based image retrieval applications,” in *Proceedings of the International Conference on Information Technology*, Bhubaneswar, India, December 2003, pp. 1–6.
- [74] A. Oliva and A. Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.
- [75] D. Gabor, “Theory of communication,” *Journal of Institute for Electrical Engineering*, vol. 93, no. 3, pp. 429–457, 1946.
- [76] A. Opelt, A. Pinz, and A. Zisserman, “A boundary fragment model for object detection,” in *Proceedings of the European Conference on Computer Vision*, Graz, Austria, May 2006, pp. 575–588.
- [77] J. Yang, Y.-G. Jiang, G. Alexander, and C.-W. Ngo, “Evaluating bag-of-visual-words representations in scene classification,” in *Proceedings of the international workshop on Workshop on multimedia information retrieval*, Augsburg, Germany, September 2007, pp. 197–206.

- [78] M. Jamieson, S. Dickinson, and S. Stevenson, “Using language to drive the perceptual grouping of local image features,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, New York, USA, June 2006, pp. 2102–2109.