Local Dependencies for near-duplicate retrieva

Cheng, Xiangang

2012

Cheng, X. (2012). Local Dependencies for near-duplicate retrieval. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/53644>

<https://doi.org/10.32657/10356/53644>

187
Hx
D/12/10

Local Dependencies for Near-Duplicate Retrieval

CHENG XIANGANG

School of Computer Engineering

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

2012

Abstract

Image representation plays an important part in many real-world multimedia applications. Over the past few years, great progresses have been achieved in this area. For example, Bag-of-Words (BOW) model is widely used in object recognition and image retrieval. However, these models do not consider the spatial information sufficiently and have severely limited descriptive ability. It is obvious the both objects and scenes have their own structures. Treating local features as independent factors will lose information, thus result in bad results. Initial researches have shown that spatial constraints play an important role in a good model for image recognition and retrieval.

In this report, we overview existing work and then present our proposed methods for image representation based on spatial relations between local features. First, we introduce the coherent phrase model which incorporates the coherency of local regions. In this model, the local regions are characterized by visual phrase of multiple descriptors instead of visual word of single descriptor. We propose two types of visual phrase to encode the coherency in feature and spatial domain respectively. The proposed model reduces the number of false matches by using this coherency and generates sparse representations of images. Second, we analyze the local dependencies among the descriptors in the Spatial-Scale Space (S-Cube). Such local dependencies in Spatial-Scale Space (S-Cube) encodes not only visual appearance but also the spatial and scale co-occurrence of them. Over the cubespace of neighboring spatial locations and multiple adjacent scales, local dependencies are exploited to form the new image representation, which is invariant to spatial transformation and scale change. Also, we extend the spatial dependency into the temporal-spatial domain. Stratification-based Keyframe Cliques (SKCs) are proposed for video description, which are more compact

and informative than frames or keyframes. Whats more, we try to integrate the local spatial information and global spatial information to build a discriminative image representation.

Acknowledgements

First, I would like to take this chance to express my sincere gratitude to my supervisor, Associate Professor Chia Liang-Tien, Clement, for his patient and altruistic assistance in my research and the preparation for this report. He gives valuable advice and inspiration on my research topic that makes research a joyful experience. Without his guidance, I could not have completed the work in this report.

Second, I would like to thank my colleagues Hu Yiqun, Gao Shenghua and Wang Huan, who have helped me with my study in the passed few years. Without their help, work in this report would be much harder for me. I am also thankful to other colleagues: Wang Zhengxiang, Ren Zhixiang, Liu Yong, Wu Min, Zhu Linhong, Zhang Yu, Zhang Juyong, Wang Danqi and Yang Shengbo for their kind support and assistance. With them, I had a good time during my PhD at CeMNet.

Third, I want to give thanks to my mentors during my internship at MSRA: Wang Jingdong, Hua Xian-sheng, and other colleagues, for their friendship and help. Many thanks to Liu Dong, Shang Lifeng, Li Darui and other interns, with whom I had one of the most memorable days at MSRA.

Last but not least, I must say thanks from bottom of my heart to my family, for their support for my PhD study.

Contents

Abstract	i
Acknowledgements	iii
List of Figures	vii
List of Tables	xi
1 Introduction	1
1.1 Background and Motivation	1
1.2 Challenges and Opportunities	2
1.3 Contributions	4
1.4 Organizations	6
2 Literature review	7
2.1 Near-duplicate Image/Video Retrieval and Detection	7
2.2 Image representation	9
2.2.1 Bag-of-Words Models	10
2.2.2 Models with Spatial Information	17
3 Coherent Phrase Model	23
3.1 Motivation	23
3.2 Coherent Phrase Model	25
3.2.1 Feature Coherent Phrase	27
3.2.2 Spatial Coherent Phrase	28
3.3 Near-duplicate Image Retrieval	31

3.3.1	Complexity Analysis	33
3.3.2	Experiment Evaluation	34
3.3.3	IND Retrieval using FCP	35
3.3.4	IND Retrieval using SCP	37
3.3.5	Time Complexity Comparison	38
3.3.6	Discussion	39
3.4	Conclusion	40
4	Exploiting Local Dependencies with Spatial-Scale Space (S-Cube)	41
4.1	Motivation	41
4.2	Spatial-Scale Space (S-Cube)	43
4.2.1	Multi-Scale Space feature extraction	43
4.2.2	Spatial-Scale Space - exploring Local Dependencies	44
4.2.3	Measure for integrating Local Dependencies	47
4.2.4	Speed-Up the retrieval with SuperNodes	48
4.3	Experiments	50
4.3.1	Experiment Settings	52
4.3.2	Scale Effects and Local Dependencies' Study	52
4.3.3	Comparison of IND retrieval results	53
4.3.4	Evaluation of online computation cost	55
4.3.5	IND guided video retrieval - Initial results	56
4.4	Conclusion	58
5	Stratification-based Keyframe Cliques for Effective and Efficient Video Representation	59
5.1	Introduction	59
5.2	Motivation	60
5.3	Related Work	63
5.4	Stratification-based Keyframe Cliques	64
5.4.1	Stratification-based keyframe cliques	64
5.4.2	Near Duplicate Keyframe (NDK) detection	65
5.4.3	Strata partition with quasi-clique	66

5.4.4	Efficient representation of SKCs	70
5.4.5	Video distance based on SKCs	71
5.4.6	Complexity Analysis	73
5.5	Experiments Evaluation	74
5.5.1	Dataset and Keyframe Representation	74
5.5.2	Evaluation of Keyframe Summarization	76
5.5.3	Evaluation of Keyframe Clique based Video Retrieval	79
5.6	Conclusions	84
6	Biologically-Inspired Spatially-Coherent Pyramid Matching	85
6.1	Introduction	85
6.2	Related Work	88
6.3	Biologically-Inspired Pyramid Matching	89
6.3.1	Subregions to describe the image structure	90
6.3.2	Maximum spatial pooling	91
6.4	Spatially-coherent Pyramid Matching	91
6.4.1	One-to-one subregion matching	92
6.4.2	Spatially-coherent region correspondence	94
6.5	Experiments	96
6.5.1	Image Near-Duplicate Retrieval	97
6.5.2	Event Classification	99
6.5.3	Webvideo Retrieval	100
6.6	Conclusion	102
7	Conclusion	104
7.1	Research Summary	104
7.2	Future Work	106
	Author's Publications	109
	Bibliography	111

List of Figures

1.1	Examples of Image Near-Duplicate varying in illumination, motion, scale and viewpoint.	3
1.2	The overview diagram of our work.	4
2.1	Framework of Bag-of-Words model.	11
2.2	Illustration of four different sampling strategies on the face image (a). The effects of the dense, random, segment, and detecting sampling strategies are shown form (b) to (e) respectively.	12
3.1	Demonstration of generating the coherent phrase model of order 3 for an image. (a) An input images I where a set of local regions are located; (b) $K = 3$ descriptors ϕ_i are extracted for every local region i ; (c) codebook of visual phrase of order 3, which is 3-dimensional array and the cardinality of each dimension is 3; (d)-(e) every local region is assigned to a visual phrase (3-tuple) in the codebook; (f) the final representation of image I , which is a sparse histogram of visual phrase frequency.	26
3.2	Two pairs of example show that SCP preserves the scale-invariant property of local region detector. The ellipses in red are the local regions output from detector in description scale s_0 , and the ellipses in yellow are the regions in expanded scales $s_1 = 1.2 * s_0$ from SCP. The corresponding regions are matched in both description scale and expanded scale.	30
3.3	An example of matching between (a) matched images and (b) unmatched images ; The first row are the results using “Bag-of-Word” of SIFT and the second row are the results using coherent phrase model (SCP). . . .	31
3.4	(a) Top- k IND retrieval performance of FCP and relevant methods on the first dataset. (b) Four labeled IND pairs (first 2 columns) from the first dataset and the most similar image to the images in the first column returned by the proposed method.	34

3.5	(a) Top- k IND retrieval performance of SCP and relevant methods on the first dataset. (b) Four labeled IND pairs (first 2 columns) from the second dataset where SCP succeeds to return them as top-1 images but SPM fails. The top-1 images returned by SPM are shown in the third column.	35
4.1	The framework of our proposed system.	43
4.2	Examples of local dependencies in both spatial and scale space. Patches with red outline are from the left IND, while those with green outlines are from the right IND. (a),(b),(c) are showing horizon, vertical, and scale invariant co-occurent neighborhood relationship between the INDs, respectively.	45
4.3	(a) A second order neighborhood system in spatial-scale space. The dark blue node is the center and it has 8 neighbors in the same scale and 9 neighbors each in both upper and lower scales respectively. (b) An example of graph representation which integrates local dependencies in the spatial-scale space. The size of a node encodes the frequency of the label associated. And the edge $(v_i, v_j) \in E$ indicates the co-occurrence frequency between the corresponding label pair.	47
4.4	Illustration of SuperNode.	50
4.5	Evaluation scale space and local dependencies on the Columbia dataset and CityU dataset. The green curve is the result by using only one scale features and the blue one is the result with multiple scale features. The red one shows the result considering both multiple scale features and local neighborhood dependencies. Note: the accuracy in (b) decreases due to multiple NDKs of one keyframe.	51
4.6	Top- k IND retrieval performance of S-Cube and relevant methods on the Columbia dataset and CityU dataset. Note: the accuracy in (b) decreases due to multiple NDKs of one keyframe.	53
4.7	Some challenging results successfully returned by our proposed approach. These near-duplicate keyframes suffers from various changes, such as illumination difference, scale variance, different editing, view point changes, motion, occlusion and so on.	55
4.8	Some failed cases returned by our proposed approach. The first row are queries(Q), and the second and third row are groundtruth near-duplicate(G) and keyframes returned by our method(R), respectively.	56
4.9	IND(s) guided video event detection (of the same event) among different videos. Video (a) is news clip from CNN. Video (b) & (c) are news clips from ABC but different date/time stamp.	57

5.1	Examples of Near-Duplicate Videos from TRECVID and Webvideo dataset.	61
5.2	The framework of our proposed system.	62
5.3	Spatial pyramid representation of keyframe of different levels. (a) level $l = 0$; (b) level $l = 1$; (c) level $l = 2$	66
5.4	Examples of overlap cliques due to different factors.	68
5.5	Stratification-based Keyframe Cliques. (a) The keyframes of one video; (b) Near-duplicate Keyframe Cliques;(c) The compact representation of NDK Cliques.	72
5.6	Video shots examples from TRECVID 2005 & 2006. (a)-(e) are the keyframes of 5 different video clips.	74
5.7	PR-curve of the NDK detections on the labeled Trecvid dataset.	77
5.8	Average near-duplicate retrieval performance comparison for different approaches over all queries. In this figure, “Temp_summ” means the method of segmentation-based temporal summarization. “Transitive_Closure” is the method based on transitive closure. “SKC_S” stands for our proposed Near-duplicate keyframe cliques based on keyframes from shot detection, while “SKC_T” stands for our proposed Near-duplicate keyframe cliques based on keyframes from dense sampling.	79
5.9	PR-curves of 24 queries for HIRACH and our proposed SKC_T method. . .	82
5.10	Average near-duplicate retrieval performance for different sampling rates of SKCs. The red curve is the result using SKCs with different sampling rates by their weights ranking, while the blue one is from random selection. . . .	83
5.11	The PR-curves for two sampling rates of SKCs with a large-scale reference set. The average precisions for sampling rate 100% and 10% are 94.66% and 92.71% respectively.	84
6.1	Examples of image pairs. Due to the scale, rotation and position variations, the image in the same class is not region-by-region correspondingly.	86
6.2	Comparison of biological-inspired spatial pyramid and the general spatial pyramid. The blue circle and red square stand for two different patterns in the image. The red numbers in BiScPM indicate the maximum-pool value for the region in the respective layer and location.	90
6.3	An illustration of how object pattern that is not aligned to subregions is handled by SPM & our BiScPM. The red box is the selected subregion. For BiScPM, maximum spatial pooling with overlapping regions will generate a better matched subregion.	92

6.4	Spatially-coherent Pyramid Matching. The subregions with colored outline are matched pairs. We see that if the matches are correct, the link between them are corresponding in the two image. Illustration is best seen in the color pdf version.	95
6.5	Top- k IND retrieval performance of our proposed method and relevant methods on the Columbia dataset and CityU dataset. Note: the accuracy in (b) sometimes decreases with the rank due to more than one NDKs for each keyframe.	96
6.6	Confusion Matrix of Princeton sports event dataset. The average accuracy is 84.79% for all the 8 categories.	101
6.7	Comparison of average Precision-Recall curves of 24 webvideo queries with different methods.	103

List of Tables

3.1	The performances of top-1 IND retrieval on two dataset for different methods	36
3.2	Average CPU time for processing a query	38
4.1	Top-1 retrieval performance comparison of S-Cube, SAPM, SPM and TPM on both Columbia Dataset and NTU Dataset. For SPM, TPM and SAPM, the first number is the performance using only one level, and the second is the highest results using multilevel with different fusion methods.	54
4.2	Comparison of overall time cost of 300 queries on the Columbia dataset	56
4.3	Comparison of overall time cost of 1953 queries on the CityU dataset .	56
5.1	Notations used throughout this algorithm of quasi-clique mining.	69
5.2	Video Queries Collected by Wu from YouTube, Google Video and Yahoo Video. $V\#$ and $ND\#$ mean number of videos and number of near-duplicate videos respectively.	75
5.3	Keyframe Summarization of TRECVID 2005 and TRECVID 2006.	78
5.4	Performance Comparison for Near-Duplicate Video Detection between our proposed method and state-of-arts methods. The average precision is of all queries over all recall levels.	81
6.1	Notation used throughout this algorithm of Hungarian Algorithm.	93
6.2	Top-1 retrieval performance comparison of our proposed method, SAPM, SPM and TPM on both Columbia Dataset. For SPM, TPM and SAPM, the first number is the performance using only one layer, and the second is the highest results using multilayer with different fusion methods.	98
6.3	Comparison of classification results on Princeton 8-sports event dataset. Note HIK adopted one scale of dense sampling points, while HIK2 adopted multi-scales of dense sampling points as well as Sobel images.	100

6.4	Performance Comparison for Near-Duplicate Video Detection between our proposed BiScPM and state-of-arts methods. The average precision is of all all queries over all recall levels.	102
-----	--	-----

Chapter 1

Introduction

1.1 Background and Motivation

Due to the popularity of digital cameras and camcorders, there is a dramatic increase of visual contents such as photos and videos in recent years. Taking Flickr¹ website for example, it is reported that the site was hosting more than 6 billion images in August and this number continues to grow steadily. Meanwhile, in the famous video-sharing website YouTube², more than two billion videos a day was serving in May 2010. In May 2011, YouTube reported in its company blog that the site was receiving more than three billion views per day. In January 2012, YouTube stated that the figure had increased to four billion videos streamed per day. It is challenging to manage the massive number of photos and videos.

Retrieval of Image Near-Duplicate (IND) [1, 2] is very useful to the filtering, retrieval and management of multimedia content. For example, the IND(s) can correlate the videos that depict the same news event from different broadcast sources and pro-

¹<http://www.flickr.com>

²<http://www.youtube.com>

vide similarity clues for recognizing visual events and searching news video clips [3]. Detecting IND(s) over internet can discover unauthorized use of private images for the application of copyright infringement detection [3]. Personal photo album can be automatically organized by grouping IND(s), which might be of different names. Detection and retrieval of IND can also facilitate traditional text-based web search. If two web pages contain any IND(s), the relevance between these two web pages should be increased.

1.2 Challenges and Opportunities

According to [1], IND is referred to multiple images that are close to the exact duplicate of one image, but may be different in scene, camera setting, photometric and digitization changes. See Fig. 1.1 for example. Specifically, the scale, viewpoint and illumination of the same scene and object(s) captured in the IND(s) can be changed by different camera settings and rendering conditions. The composition of multiple objects can be different in the IND(s) due to some editing operations. What's more, the near-duplicates usually cover only a small ratio of the images/videos in a collection/websites. Finding the near-duplicate images/video has to overcome different kinds of noises and wrong matches. Thus, to well detect the near-duplicates, the algorithm has to be robust and discriminative to tackle the challenges mentioned above.

Image representation plays a key role in object recognition and image retrieval, which transfers an abstract image with a representative feature vector. Usually, a feature vector explores certain type of image characteristics, such as color, edge, texture and so on. Recently, local features have been shown to be well suited in many applications as they are robust to occlusion, background clutter and other content changes. A patch-based description or representation of an image can be obtained by processing the

Figure 1.1: Examples of Image Near-Duplicate varying in illumination, motion, scale and viewpoint.

image with one of the specialized scale or affine invariant salient region detectors that have been introduced over the last decade [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16]. Based on local features, several image representations have been proposed. Bag-of-Words (BoW) based model [17, 15, 18, 19, 20, 21] is one of them, which achieves a promising performance for object recognition and image retrieval. BOW models usually represent each image as a histogram of occurrence of the codewords in a codebook. However, because these methods disregard all information about the spatial layout of the features, they have severely limited descriptive ability.

The BoW provide us an important clue for near-duplicate retrieval. However, we go further beyond it. It is obvious the both objects and scenes have their own structures. Treating local features as independent factors will lose information. Incorporating the local dependencies of local features will greatly enrich the semantics in the image, thus improve the representative power.

Figure 1.2: The overview diagram of our work.

1.3 Contributions

The goal of this thesis is to explore local dependencies for high level image and video representations. In this dissertation, we try to tackle this issue from different aspects. Fig. 1.2 shows the overview diagram of our work. The main contributions of this thesis are:

1. We introduce the coherent phrase model which incorporates the coherency of local regions to reduce the quantization error of the bag-of-words (BoW) model. In this model, local regions are characterized by visual phrase of multiple descriptors instead of visual word of single descriptor. Two types of visual phrases are proposed to encode the coherency in feature and spatial domain, respectively. The proposed model reduces the number of false matches by using this coherency and generates sparse representations of images. Compared to other method, the local coherencies among multiple descriptors of every region improve the

performance and preserve the efficiency for IND retrieval.

2. We also analyze the local dependencies among the descriptors in the Spatial-Scale Space (S-Cube). Such local dependencies in spatial-scale space(S-Cube) encodes not only visual appearance but also the spatial and scale co-occurrence of them. The local dependencies are exploited over the cube-space of neighboring spatial locations and multiple adjacent scales to form the new image representation, which is invariant to spatial transformation and scale change. To speed up the retrieval process, the SuperNodes are built to incorporate the neighbor information.
3. What's more, in the spatial-temporal domain, we propose a strategy to generate Stratification-based Keyframe Cliques (SKCs) for video description, which are more compact and informative than frames or keyframes. The new representations are scalable for different retrieval tasks due to the ranking of SKCs. To further accelerate the retrieval speed, only top SKCs will be used, meanwhile, the searching results are still satisfactory.
4. Finally, we present a method of max-pooling spatially-coherent pyramid matching (MpScPM). Higher-layer representations are generated from lower-layer subregions, by a biologically-inspired max pooling strategy. Second, instead of reshaping the pyramid representation into a vector (used in generic SPM), the layer and location information of each subregion are kept and weak geometrical correspondences between matched subregions are explored to enhance our pyramid matching method. To enhance the possibility of finding the best matches at different scales and locations, cross-layer region similarities are computed, while the correspondences (either spatial neighbors or adjacent layers) are also incorporated.

1.4 Organizations

The thesis is organized as follows: Chapter 2 gives the related work on widely-used image representation models and their extensions with spatial techniques. Then we introduce our proposed local dependencies based on spatial domain in the next two chapters. Specifically, Chapter 3 introduces the coherent phrase model which incorporates the coherency of local regions to reduce the quantization error of the bag-of-words (BoW) model. Chapter 4 analyzes the local dependencies among the descriptors in the Spatial-Scale Space (S-Cube). Such local dependencies in spatial-scale space(S-Cube) encodes not only visual appearance but also the spatial and scale co-occurrence of them. The local dependencies are exploited over the cube-space of neighboring spatial locations and multiple adjacent scales to form the new image representation, which is invariant to spatial transformation and scale change. After that, we extend the local dependencies from spatial domain to temporal domain in Chapter 5. A strategy to generate Stratification-based Keyframe Cliques (SKCs) is proposed for video description, which are more compact and informative than frames or keyframes. Chapter 6 tries to incorporate both local and global spatial information for image representation. We present a method of max-pooling spatially-coherent pyramid matching (MpScPM). Higher-layer representations are generated from lower-layer subregions, by a biologically-inspired max pooling strategy. Second, instead of reshaping the pyramid representation into a vector (used in generic SPM), the layer and location information of each subregion are kept and weak geometrical correspondences between matched subregions are explored to enhance our pyramid matching method. Finally, we summarize the thesis and conclude with some directions for future research.

Chapter 2

Literature review

2.1 Near-duplicate Image/Video Retrieval and Detection

Near-duplicate image retrieval and detection is challenging due to the variances in scene, camera setting, photometric and digitization. At the beginning, researches about exact duplicate and copy detection are mainly based on image representation using global features. Global features such as color moment and color histogram [22, 1], although efficient, are sensitive to occlusions and illumination changes. Later, some researchers have proposed to use invariant local features to detect and retrieve INDs. Zhang and Shi [1] proposed to identify IND using a stochastic attributed relational graph (ARG) matching. They learnt a distribution-based similarity from the spatial relation among local interest points for matching ARGs. However, the matching speed of this method is slow and the parameters can only be tuned heuristically. Different from [1], Ke et al. [23] represented each image as a set of local covariant regions each of which is characterized by PCA-SIFT descriptor. Locality sensitive hashing (LSH)

is proposed to design an efficient index structure for fact point set matching. Without geometry verification, the robustness of this technique cannot be guaranteed for partial matching due to the clutter background. Recently, Zhao et al. [2] extended the matching of local point set by introducing one-to-one symmetric property for matching and LIP-IS index structure for fast approximate search. Although it is interesting to learn IND pattern from the histogram of matching orientation, the method proposed in [2] is not invariant to rotation and requires explicitly calculating point-to-point matching when comparing two images.

Generic image categorization approaches using “Bag-of-Words” model is also related to IND detection and retrieval. Under this model, images are treated as documents by assigning descriptors of local covariant regions to “visual words”. Each image is then represented by a histogram of word frequency. The most critical problem of such methods is that the ambiguous visual words will introduce large number of false matches when each region is matched independent to others. Several methods have been proposed to improve this problem by capturing the spatial arrange of visual words. Lazebnik et al. [24] extended the Pyramid Matching Kernel (PMK) [25] by incorporating the spatial information of location regions. Two images are partitioned into increasingly fine sub-regions and PMK is used to compare corresponding sub-regions. This method implicitly assumes the correspondences between sub-regions, which is not scale or rotation invariant. Also by utilizing spatial information of local regions, Savarese et al. [26] used correlogram to measure the distribution of the proximities between all pairs of visual words and used for category classification, but the method is not scale invariant.

Generally speaking, for near-duplicate video retrieval there are two strategies. The earlier methods rely on the correlation of temporal frames. Naphade et al.[27] proposed a sequence matching algorithm for video clips, by considering linearized histograms of

the DCT frames and its variation of the histogram intersection. However, this method is very sensitive to signal variations. Hampapur et al.[28] proposed a sequence-matching technique by using global signatures of the video, such as motion, color or distribution of intensities. The second strategy is based on keyframe matching. Li et al.[29] adopted a compact binary signature involving color histograms for keyframe matching. The method is efficient, however, the effectiveness is limited due to their lack of robustness with regard to editing or shifting. Hsu et al.[30] proposed a partition based approach, in which color information of each partition is obtained and the similarity of two images is measured by the similarity of all the sub-images. Recently, Wu et al. [31] proposed a hierarchical approach to detect near-duplicate videos, which combined global signatures and matching pairs between local features. The main advantage of keyframe-based video search methods is that they are less dependent on the time duration in a video and robust to video editing operations. Whereas, it suffers from the time-consuming problem of matching all keyframes, especially for those methods relying on local-points matching. To solve this problem, Tan [32] proposed alignment-based matching to eliminate exhaustive keyframe pairs matching. However, existing near-duplicate keyframes in the same video may not be time-aligned and there exist one-to-many, many-to-many matches, which will make alignment based method less effective.

2.2 Image representation

Image representation plays a key role in object recognition and image retrieval, which transfers an abstract image with a representative feature vector. Usually, a feature vector explores certain type of image characteristics, such as color, edge, texture and so on. Recently, several sophisticated models for object recognition and image retrieval

have been proposed. For example, part-based model [33, 14, 34, 35, 36, 37, 38, 39] is a widely used model. It represents an object as a spatial layout of multiple parts, where the deformable configuration is characterized by spring-like connections between parts [40]. such constellation model is learned by using a maximum likelihood unsupervised learning algorithm [33]. Fergus et al [41] simplified full constellation model with star model for reducing complexity while maintaining a high accuracy. The disadvantage of existing part-based model is that it heavily depends on the representations of each part. Meanwhile, Bag-of-Words (BOW) [19] is one of the most widely used models for image representation and still of much concern in current research. Related to my research, in this chapter I reviews existing work about three widely-used models: Bag-of-Words (BOW) [19], which assumes local features are independent from each other; global spatial BoW models, which explore the global insistence of an object or scene image, local spatial BoW models, which capture local correspondence between interest points.

2.2.1 Bag-of-Words Models

Local features [5, 42] have been shown to be well suited to retrieval and recognition as well as to many other applications. Based on local features, Bag-of-Words (BOW) [19] is an effective approach for generic image categorization and retrieval. Under this model, images are treated as documents by assigning descriptors of local regions to “visual words”. Each image is then represented by a histogram over visual words.

Originally, BOW models are used in the information retrieval domain [43, 44]. Its basic idea is to represent each document as a vector which reflects the distribution of the words in the dictionary. This representation simply assumes that the words in the dictionary are independent, which means that the occurrence of one word has no

relationship with other words and it can appear at any place in the document.

Inspired by the success of BOW models, many researchers managed to apply this representation to computer vision and multimedia. Similar to information retrieval domain, there is also an assumption of independence in computer vision. Exactly, the patches in each image are assumed independent from each other. Then the images are represented as a histogram of visual words. So far, BOW is used to model images for different purposes, such as scene classification [15, 18], image retrieval [19, 20], object recognition [17, 19] etc.

Figure 2.1: Framework of Bag-of-Words model.

Generally speaking, there are five steps in this framework of BOW model, see Fig.2.1. First, interesting regions of the training images are extracted in the detection module. Then these regions are described as local feature vectors by descriptors in the description module. Third, the cluster module classifies these local feature vectors

into different clusters, whose centers are taken as codewords to generate a codebook. After this, each local feature vector is assigned to a visual word, which is the closest in the codebook. By doing this, each image can be represented as a histogram of visual words.

Interesting region detection and selection

For local features, the first important step is to detect or select certain regions. Interesting patches/regions are extracted from images, which may be important for object categorization. An image patch is a small part of an image where each pixel is connected with others. Generally speaking, there are four types of sampling strategies: dense sampling (e.g. [24]), random sampling (e.g. [15]), segment sampling (e.g. [45]), and detecting sampling (e.g. [46]). Fig.2.2 illustrates these three strategies.

Figure 2.2: Illustration of four different sampling strategies on the face image (a). The effects of the dense, random, segment, and detecting sampling strategies are shown from (b) to (e) respectively.

In the dense sampling, images are usually divided into grids with the same-size cells. The advantages of this sampling strategy are that (1) it can control the number, centers and scales of the patches, and (2) it can utilize the information of each image sufficiently because the patches cover the whole image. However, since there is a fixed grid structure in the dense sampling, it cannot locate each characteristic patch properly

in the images which can distinguish the foreground objects from others with different categories, e.g. it cannot locate the nose in the face image properly. This problem can be relaxed by changing the number of patches efficiently.

In the random sampling, the centers and scales of the patches are decided randomly. The advantages of this sampling strategy are that (1) it can control the number of the patches, and (2) it can probably locate some characteristic patches in the images. However, it cannot control the centers and scales of the patches, and it can hardly utilize the information from all the parts of each image. Increasing the number of patches efficiently may relax this problem. In the segment sampling, some segmentation techniques (e.g. [47]) are used to extract patches from the images.

Comparing to the dense or random sampling strategy, which cannot control the positions and scales of the patches, some interesting point or region detectors are employed in the detecting strategy, to locate the centers and scales of the patches in each image. The basic difference between interesting point detectors and region detectors is that interesting point detectors try to locate the points first which satisfy some mathematical conditions and then decide the scales of the patches based on these points, while region detectors try to locate the patches directly which also satisfy some mathematical conditions. At present, quite a few interesting point detectors or region detectors have been proposed, such as edge-laplace detector, harris-laplace detector, hessian-laplace detector, harris-affine detector, hessian-affine detector [48], maximally stable extremal regions(MSER) [4], IBR&EBR [49], salient regions detector [50]. A comparison of some of these detectors can be found in [42]. From their experiments, we see that it seems that MSER is usually best under different conditions. However, for different tasks on the different datasets, the best detector is usually different. Obviously, the advantage of such strategy is that the points located keep certain characteristics and are more stable against scale and rotation changes. However, it can only select a certain number

of points and cannot utilize the information from all the parts of the images.

Nowak et al. [51] did some research on the effect of different sampling strategies on the performances of image classification. It shows that the single most important factor governing performance is the number of patches sampled from the test image and ultimately interesting point detectors cannot provide enough patches to compete, so the random sampling strategy gives equal or better classifiers than the detecting strategy. [15] compares the dense (grid), random, and detecting sampling strategies in their experiments, and addresses that the dense sampling, which yields the most number of patches, is the best.

Feature description

Descriptor is a feature vector that describes the characteristics of each patch. So far, many types of descriptors have been proposed, such as Scale Invariant Feature Transform (SIFT) [5], PCA-SIFT [6], Gradient Location-Orientation Histogram (GLOH) [7], Speeded Up Robust Features (SURF) [8], spin images [9], shape context [10], filter banks (e.g. [11, 12]), color moments [13], color histograms, pixel intensity vectors (e.g. [14, 15, 16], Histogram of Gradients(HOG)[52] etc.

Some research works have been done for comparison of the performances of different descriptors. In [7], the performances of some different descriptors are compared, and the conclusions are drawn that (1) in most of the tests, GLOH obtains the best results, closely followed by SIFT, (2) shape context also shows a high performance when edges are reliable, and (3) it seems that robust region-based descriptors perform better than point-wise descriptors. [6] shows that PCA-SIFT is better than SIFT, but this is not the case in [7]. And the experiments in [8] demonstrate that SURF seems better than SIFT. However, currently, many researchers still favor to adopt the SIFT descriptor in

their experiment.

Clustering Approach

It is very complex and time-consuming if we use the high-dimensional descriptors directly. Varying in cardinality and lacking meaningful ordering of descriptors result in difficulty of finding an acceptable model to represent the whole image. To address those problems, clustering techniques are used in “Bag-of-Words” method [19]. Clustering aims to perform vector quantization (dimension reduction) to represent each descriptor using the codewords in a codebook. Some typical clustering approaches are employed to form the codebooks, such as k-means [53, 54], hierarchical k-means [55, 56, 57], approximate k-means [58], etc.

The k-means algorithm tries to cluster n feature vectors (descriptors) in a vector space into k ($k < n$) clusters to form a codebook, in which each word is the center of natural clusters of descriptors. by minimizing the total intra-cluster variance, or the square error function:

$$V = \sum_{i=1}^k \sum_{x_j \in S_i} (x_j - \mu_i)^2 \quad (2.1)$$

where there are k clusters S_i ($i = 1, 2, \dots, k$), and μ_i is the centroid (mean point) of all the points $x_j \in S_i$. K-means starts by partitioning the input points into k initial clusters, either at random or using some heuristic data. It then calculates the center of each cluster. It constructs a new partition by associating each point with the closest center. Then the centers are recalculated for the new clusters, and the algorithm repeated by alternate application of these two steps until convergence, which is obtained when the points no longer switch clusters (or alternatively the centers are no longer changed).

K-means is quite popular for vector quantization for its simplicity. However, it still suffers several drawbacks. First, the cluster number k need to be pre-defined

first, and an inappropriate choice of k may yield poor results. Second, clusters from these methods are ranged orderless and the similarity between different clusters is not considered in the clustering method. In addition, in k -means the points lying far from any of the centers can significantly distort the position of the center that they are assigned to, which results in the inaccurate estimation of the center.

Considering these, in [59] the authors proposed an improved version of k -means with mean-shift [60]. Their algorithm produces an ordered list of centers, with the quantization rule that patches are assigned to the first center in the list that lies within a fixed radius of them, or left unlabeled if there is no such center.

The hierarchical k -means algorithm organizes all the centers of clusters in a tree structure. The algorithm divides the data recursively into clusters. In each iteration (each node of the tree), k -means is utilized by dividing the data belonging to the node into k subsets. Then, each subset is divided again into k subsets using k -means. The recursion terminates when the data is divided into single data points or a stop criterion is reached. One difference between k -means and hierarchical k -means is that k -means minimizes the total distortion between the data points and their assigned closest cluster centers, while hierarchical k -means minimizes the distortion only locally at each node and in general this does not guarantee a minimization of the total distortion.

Besides k -means, some other approaches have been proposed for the codebook generation. For instance, [61] proposed Extremely Randomized Clustering Forests-ensembles of randomly created clustering trees; [62] employed a graphical model-Latent Dirichlet Allocation (LDA) [63].

In addition to the vector quantization approaches, some codebook generation strategies are also proposed. For example, in [64] the authors proposed a strategy to generate an optimally compact visual dictionary-universal codebook-by pair-wise merging of vi-

sual words, optimized by a statistical measure of discrimination, from an initially large dictionary. The final visual words are described by GMMs. And furthermore, in [65] the authors proposed another strategy to describe each image based on a universal codebook, which is generated using all the training data, and class codebooks obtained through the adaptation of the universal codebook using class-specific data.

Summary

“Bag-of-Word” (BOW) models[19] have shown remarkable performance in recognition and classification. Under this model, images are treated as documents by assigning descriptors of local invariant regions to “visual words”. Each image is then represented by a histogram of word frequency. To some extent, this model is robust and insensitive to scale and illumination change but BOW suffers from the lack of any available spatial information, which severely limits its descriptive ability.

2.2.2 Models with Spatial Information

Bag of word models have severely limited descriptive ability, due to disregarding all the spatial layout information of the features. In particular, they are incapable of capturing shape or of segmenting an object from its background. In this section, we review two strategies to incorporate spatial information. To be exact, one is global spatial information based BOW and the other is local spatial information.

Models with Global Spatial Information

Before we survey the extended BOW model, which incorporates spatial information, first we have a brief review about part-based models, which represent an object as a spatial layout of multiple parts. The deformable configuration is characterized by

spring-like connections between parts [40] . such constellation model is learned by using a maximum likelihood unsupervised learning algorithm [33]. Fergus et al [41] simplified full constellation model with star model for reducing complexity while maintaining a high accuracy. The disadvantage of existing part-based model is that it heavily depends on the representations of each part. In terms of computational complexity, the constellation model is very expensive. Suppose N is the number of feature detections in the image, and P is the number of parts in the object model. Generally speaking, the structure graphs of the model can be: fully connected, star structure, and tree structure; and their corresponding computational complexity are: $O(NP)$, $O(N^2P)$, and $O(NP)$.

For BOW model, it is reasonable to overcome its limitations with the combination of part-based model. Unfortunately, the efficiency of BOW model will be lost if doing so. To make a trade-off, a kernel-based recognition method which includes the spatial layout based on approximate global geometric correspondence was introduced by [24]. This technique works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region using the pyramid matching scheme of Grauman and Darrell [25]. It involves repeatedly subdividing the image and computing histograms of local features at increasingly fine resolutions. This simple operation suffices to significantly improve performance over a basic bag-of-features representation. However, the Spatial Pyramid Matching (SPM) method implicitly assumes matching occurs between corresponding sub-regions in the same layer and location. It is therefore not scale or rotation invariant, as well as sensitive to the change in position. What is more, all the BOW based methods can not distinguish between objects and background and they are sensitive to background clutter.

A more efficient approach [66] is to augment a basic bag-of-features representation with “semi-local parts” that combine neighboring features by validating their affine

relations. Characteristic patterns formed from scale or affine-invariant patches linked by semi-local spatial relations are used to describe salient object parts. These invariant semi-local parts are geometrically stable configurations of multiple invariant regions, found by the Laplacian keypoint detector or by the shape detector [67] and are more distinctive than individual features. Later [38] framed their semi-local parts into a maximum-entropy approach to learn the posterior distribution of the class label given the occurrences of parts from the dictionary in the training set and performed well on textures as well as objects. However, as reported by [24], existing implementations of this idea [66, 38] have yielded inconclusive results.

Models with Local Spatial Information

Markov models for the spatial organization of image descriptors. Previous work on texture recognition [70] introduced a method that simultaneously performs classification and segmentation. A generative model describes the distribution of the affine-invariant descriptors, along with co-occurrence statistics for nearby patches. At recognition time, initial probabilities computed from the generative model are refined using a relaxation step that incorporates co-occurrence statistics learned at modeling time.

Global co-occurrence statistics do not explicitly model the local dependencies between neighboring descriptors, and recognition results can be further improved by modeling these dependencies. Markov Random Fields (MRF) was selected as an appropriate statistically-based framework for this task. They provide parametric models whose parameters have a natural interpretation and they can be adjusted to incorporate a priori knowledge with respect to interaction strengths. Using them requires nontrivial parameter estimation, for example, estimation based on the mean field principle. The particularities of the application are the high-dimensionality of the feature vectors (typically more than 100 dimensional) and the irregularity of the sites at which

they are observed. Very few practical optimization techniques are available for such tasks. They are usually very sensitive to initialization and to the parameters of the approach. By combining an MRF estimation procedure with a dimensionality reduction technique and recognition rates can be improved and that promising results can be obtained using a general statistical formalism. However, their work focused only on texture recognition [71, 72], and other contexts such as object recognition were not included.

Discriminative field models were presented to capture spatial interactions in images in a discriminative framework based on the concept of Conditional Random Fields proposed by Lafferty et al [73]. The discriminative fields offer several advantages over the Markov Random Fields (MRFs) popularly used in computer vision. First, they allow to capture arbitrary dependencies in the observed data by relaxing the restrictive assumption of conditional independence generally made in MRFs for tractability. Second, the interaction in labels in discriminative fields is based on the observed data, instead of being fixed a priori as in MRFs. This is critical to incorporate different types of context in images within a single framework. Finally, the discriminative fields derive their classification power by exploiting probabilistic discriminative models instead of the generative models used in MRFs.

Recently, several methods have been proposed to improve on BOW model by capturing the spatial layout of visual words. Utilizing spatial information of local regions, Savarese et al. [26] used correlagram to measure the distribution of the proximities between all pairs of visual words. To make the traditional bags of features approaches geometrically unconstrained, Ling [21] introduced the ‘proximity distribution kernel (PDK) that naturally combines local geometric as well as photometric information from images. PDK captures rank information between visual words, which is more reliable and sparse. Insensitivity to occlusions is achieved through the use of a min

functional in the definition of PDK, and insensitivity to the discontinuities in the proximity relations is achieved by using the probability distribution in place of a (density) histogram.

Co-occurrence information has drawn great interest in recent research. Li et al. [74] proposed a general framework called Markov stationary features (MSF) to extend histogram based features. MSF characterizes the spatial co-occurrence of histogram patterns by Markov chain models, and finally yields a compact feature representation through Markov stationary analysis. The intra-histogram-bin and inter-histogram-bin relationships are encoded into histograms, where the initial and stationary distributions of the Markov chain model are combined to form the so-called Markov stationary features. To improve MSF, Ni et al. [75] proposed the concept of homogeneity-aware Markov stationary features (HMSF) and contextualizing histogram, which encodes the spatial contexts as local homogeneity distributions and produces the contextualized histograms by convoluting local homogeneity distributions with the histogram-bin. The above methods proved that the co-occurrence information can improve the performance when compared to the standard BOW histogram. However, their methods are still limited in many aspects. For example, they only explore two kind of co-occurrence information, that is, the intra-bin information and the extra-bin information, while different visual words co-occurrence information does not differ in the extra-bin. What's more, only the spatial distribution is considered and the neighbors are treated equally so that the hard assignment (0 or 1) is adopted.

Another approach which adds spatial relations to a bag-of-features representation is spatial weighting for bag-of-features [76]. The idea is to exploit spatial relations between features based on object boundaries provided during supervised training. Each feature of a test image votes for the aligned object shapes of the corresponding features in the training images. The weights of features which agree on the position and

shape of the object are boosted and the weights of background features are suppressed. Unlike most other localization methods, this approach do not require any hypothesis parameter space to be defined. Instead, it directly generates, evaluates and clusters shape masks.

Recent works on the indexing and geometric verification are also introduced to improve BoW model, which are used to tackle the issue of false matches caused by the vector quantization error. Jégou [68] etc. proposed visual verification, namely Hamming embedding (HE) and weak geometric constraint (WGC) to prune these false visual word matches. In [69], a method is proposed to learn discretized local geometry representation based on minimization of average reprojection error in the space of ellipses. The representation requires only 24 bits per feature without drop in performance. However, the assumption that image scaling must be within a certain range restricts its feasibility in practice.

In all, we conclude that incorporating the spatial information is of great help for image representation, though the approaches highlighted above have their own weaknesses. In the next few chapters, we try to present our algorithms to model more discriminative image representations.

Chapter 3

Coherent Phrase Model

3.1 Motivation

We introduce the coherent phrase model which incorporates the coherency of local regions to reduce the quantization error of the bag-of-words (BoW) model. In this model, local regions are characterized by visual phrase of multiple descriptors instead of visual word of single descriptor. We propose two types of visual phrase to encode the coherency in feature and spatial domain, respectively. The proposed model reduces the number of false matches by using this coherency and generates sparse representations of images. Compared to other method, the local coherencies among multiple descriptors of every region improve the performance and preserve the efficiency for IND retrieval. The proposed method is evaluated on several benchmark datasets for IND retrieval. Compared to the state-of-the-art methods, our proposed model has been shown to significantly improve the accuracy of IND retrieval while maintaining the efficiency of the standard bag-of-words model. The proposed method can be integrated with other extensions of BoW.

In this chapter, we propose an efficient and effective coherent phrase model. This model uses **phrase** of multiple descriptors instead of **word** of single descriptor to characterize every local region to enforce local coherency . Specifically, we propose two types of phrase:

- Feature Coherent Phrase (FCP): every local region is characterized by multiple descriptors of different features e.g. SIFT [5] and SPIN [77]. The match of two regions requires the coherency across different types of features.
- Spatial Coherent Phrase (SCP): multiple descriptors of a single feature are generated from the neighborhood of different scales of every local region. The match of two regions requires the coherency across different spatial neighborhoods.

The proposed coherent phrase model has several advantages. First, it improves the *effectiveness* of “Bag-of-Word” model by introducing visual phrase, which enforces the coherency constraint across multiple descriptors to remove false matches. Second, it provides an *efficient* framework to explore spatial/appearance relationship of local regions. The efficiency lies in no need to construct the codebook of visual phrase and the matching between two images. Multiple descriptors of a local region can be assigned to a visual phrase by assigning every descriptor to a visual word separately. The maximum number of non-zero phrase is kept the same as the number of local regions such that the representation of image is sparse and the distance is fast to compute. Through the evaluation on the keyframes of TRECVID [78], the proposed two models are shown to outperform standard “Bag-of-Word” model and existing state-of-the-art algorithms [24, 2].

The rest of the chapter is organized as follows. Section 3.2 gives the detail description of the proposed coherent phrase model including the design of two types of phrase and the generation of the codebook of visual phrase. The experiment results

on two datasets from the keyframes of TRECVID are reported and compared with other methods in Section 3.3.2. Finally, we conclude the contributions of this chapter in Section 3.4.

3.2 Coherent Phrase Model

“Bag-of-Word” representation, although efficient, ignores the spatial information [26] and results in the ambiguity of visual words [79]. In this section, we propose a new coherent phrase model which is an image representation to enhance the standard “Bag-of-Word” model. Given an image $I : \mathbb{R}^2 \rightarrow \mathbb{R}^3$, M local regions can be located by using some detector [21] or using dense sampling [24]. For every local region i , a combination of K descriptors ϕ_{ik} are extracted from the local region to characterize its appearances.

$$\phi_i = \{\phi_{i1}, \phi_{i2}, \dots, \phi_{iK}\}. \quad (3.1)$$

Each ϕ_i is mapped to a K -tuple of visual words. We call this K -tuple as a **Visual Phrase** v^p ,

$$v^p = \{v_k^p | k \in [1, 2, \dots, K]\} \quad (3.2)$$

where each v_k^p indicates a visual word. Every ϕ_{ik} , $k \in [1, 2, \dots, K]$ belongs to one of the visual words. K is denoted as the order of visual phrase. By assigning every ϕ_i to the corresponding visual phrase, we obtain the coherent phrase model of an image, which is a histogram of visual phrase frequency. Figure 3.1 illustrates the process of generating this representation where the order of visual phrase is 3.

The differential property of the coherent phrase model is that under this model, every local region is characterized by *multiple* descriptors. Multiple descriptors of every local region describe different aspects of the appearance. Hence, we can incorporate the meaningful coherency within multiple descriptors of every point. Specifically, two local

Figure 3.1: Demonstration of generating the coherent phrase model of order 3 for an image. (a) An input images I where a set of local regions are located; (b) $K = 3$ descriptors ϕ_i are extracted for every local region i ; (c) codebook of visual phrase of order 3, which is 3-dimensional array and the cardinality of each dimension is 3; (d)-(e) every local region is assigned to a visual phrase (3-tuple) in the codebook; (f) the final representation of image I , which is a sparse histogram of visual phrase frequency.

regions are recognized as a matched pair only when all of K descriptors are matched to each other.

$$\phi_i \doteq \phi_j \iff \bigcap_{k=1}^K (\phi_{ik} \doteq \phi_{jk}), \forall i, j \in [1, \dots, M] \quad (3.3)$$

where $\doteq$ denotes the match indicator. Since every pair of descriptors (ϕ_{ik} and ϕ_{jk}) match to each other, they belong to the same visual word v_k^p . ϕ_i and ϕ_j belong to the same visual phrase v^p . Thus we can find the matching of two point efficiently by assigning them to visual phrase. Different types of coherency can be enforced by designing different types of visual phrase, where K descriptors are generated in different ways. In the following subsections, we propose two types of visual phrase, where both

feature and spatial coherency can be enforced, respectively.

3.2.1 Feature Coherent Phrase

Multiple types of local descriptors [77, 5, 9] have been proposed in the literature to represent local image region. According to the study in [7], different descriptors are compromise because they represent different features about the local region. For example, SIFT [5] and its variants e.g. [6] records the statistics of gradient orientation, SPIN [9] encodes the distribution of intensity at different distances from a center, and Shape Context [77] describes the information of shape.

If we extract multiple types of local descriptors from a local region, we can construct the visual phrase consisting of different types of descriptors, which is called **feature coherent phrase (FCP)**. For example, we can generate FCP of order 2 as follows

$$\phi_i(2) = \{\phi_{i1} = D_{sift}(R_i), \phi_{i2} = D_{spin}(R_i)\} \quad (3.4)$$

where R_i is the corresponding region around point i , $D_{sift}(R_i)$ and $D_{spin}(R_i)$ are the SIFT and SPIN descriptors extracted from R_i respectively. FCP enforces the coherency across different types of feature between two matched local region. When matching two FCPs using Equation (3.3), it is required that every feature of two local regions to be matched for matching these two regions.

For descriptors of k^{th} type used in FCP, we perform vector-quantization to obtain V_k visual words of k^{th} type $VV_k = \{v_1, v_2, \dots, v_{V_k}\}$. K codebook of visual words of K different types are obtained. Notice that the codebook size V_k can be different for different types of descriptor. For every local region, we map K descriptors to FCP (K -tuple of visual words) by every descriptor to some visual word in the corresponding codebook of same type, separately.

3.2.2 Spatial Coherent Phrase

Every local region is associated with a scale, which is either provided by the detector or fixed for dense sampling. Local descriptor is commonly extracted from a region of such scale. We denote this scale as *description scale*. According to the assumption of spatial coherency [80], the neighborhood of two matched points is very possible to match to each other. This motivates the design of **spatial coherent phrase (SCP)**.

Given a local region i as well as the associated scale s_0 , we expand original scale s_0 to multiple expanded scales as

$$s_k = C \cdot s_{k-1}, \quad k \in [1, 2, \dots, K] \quad (3.5)$$

where the constant C is the expansion ratio between two consecutive scales. K descriptors are extracted from the expanded region of these K scales using only one type of descriptor

$$\phi_i = \{\phi_{ik} = D(R_{i,s_k}) | k \in [1, 2, \dots, K]\} \quad (3.6)$$

where R_{i,s_k} denotes the local region of scale s_k centered at the same center of the local region i and $D(R_{i,s_k})$ denotes the descriptor (e.g. SIFT) extracted from R_{i,s_k} . SCP enforces the coherency across different spatial neighborhoods of two matched points. Two local regions are matched only when their spatial neighborhoods of K expanded scales centered at the same center are consistently matched.

When selecting the order (K) of SCP, there are two factors to be considered. Given the codebook size of single visual word, the order K of SCP controls the codebook size of visual phrase. A suitable K needs to be selected to generate desired number of visual phrase. Given the expansion ratio C , the order K of SCP controls the spatial context of every local region. Because spatial coherency is only applicable in the local context, a suitable K needs to be selected to constrain the size of local context. Because the codebook size of each dimension is 500 in our experiment, we use only $K = 2$, which

generates a codebook of $500^2 = 250000$ visual phrases. By reducing the codebook size of each dimension, e.g., from 500 to 50, we can design SCP of higher order. We will conduct an experiment to analyze how the order of SCP affects the IND retrieval performance in the experiment section.

Note that SCP strictly utilizes the description scale from the detector and expands it accordingly using a fixed expansion ratio. SCP preserves the chance of two local regions to get matched as the same as from the detector. This is because the original description scale output from the detector is scale invariant, which allows for repeatedly detecting two matched regions in different scales. By expanding those two description scales using a fixed expansion factor, two corresponding regions in the new expansion scales are the matched region, and hence, their descriptors in two corresponding expansion scales can be matched. SCP uses the characteristic scales of the regions output from detector as the reference for expanding scales using a fixed expansion factor. It achieves the same invariant ability to scale transformation as the detector. Fig. 3.2 illustrates this property of SCP using two pairs of images. We can see that the same regions appear at different scales in two images because they undergo a scale transformation. The detector output these two scales (red) as the characteristic scales of them, which allow for detecting the matched region in two images. By expanding the two characteristic scales using a fixed expansion factor (e.g., 1.2), we obtain two matched regions in corresponding expanded scales (yellow) because the expansion uses the characteristic scales from detected region as the reference. At every corresponding expanded scale, the obtained regions can be matched in the same way. Hence, SCP preserves the scale-invariant property of the detector and allows for matching the regions undergoing different transformation. Similar to FCP, we perform k -means clustering on all the available descriptors of one type used in SCP. The obtained V_k visual words $VV_k = \{v_1, v_2, \dots, v_{V_k}\}$ form a codebook of single type of descriptor. A local region is mapped

to a SCP (K -tuple of visual words of the same type) by assigning K descriptors to some visual word in the single codebook, separately.

Figure 3.2: Two pairs of example show that SCP preserves the scale-invariant property of local region detector. The ellipses in red are the local regions output from detector in description scale s_0 , and the ellipses in yellow are the regions in expanded scales $s_1 = 1.2 * s_0$ from SCP. The corresponding regions are matched in both description scale and expanded scale.

The proposed coherent phrase representation provides a general framework to incorporate coherency across multiple descriptors of every local region. The proposed FCP introduces the feature coherency across different types of descriptors and SCP introduces the spatial coherency across neighborhoods of different scales. Both of these coherency can effectively reduce the number of false matches due to the ambiguity of single local descriptor and the errors caused by vector-quantization. Figure 3.3 shows an example of matching using “Bag-of-Word” and coherent phrase model (SCP of order 2) representations. We can see all false matches (excepted one) using visual word are removed when using visual phrase and most of the true matches are preserved. Suppose the probability that a false match occurs using “Bag-of-Word” model is p , the probability that a false match occurs using the coherent phrase model of order K is

Figure 3.3: An example of matching between (a) matched images and (b) unmatched images ; The first row are the results using “Bag-of-Word” of SIFT and the second row are the results using coherent phrase model (SCP).

only p^K , which is much smaller than p . For example, when the probability of occurring a false match is 0.3 for “Bag-of-Word”, the probability of occurring a false match using the coherent phrase model of order 2 is only $0.3^2 = 0.09$. Besides FCP and SCP, it is possible to design other visual phrase to incorporate other forms of coherency. One example is to combine the proposed FCP and SCP into a hybrid phrase which considers both feature and spatial coherency.

3.3 Near-duplicate Image Retrieval

We use a similar framework to [19, 81] for IND retrieval. It consists of two parts: the process for database setup and the process for handling input query. The process for database setup includes:

1. Locate local regions, extract descriptors from every local region for all images in the database. Here we use dense sampling [24] to extract descriptors on an overlapped 8×8 grid to cover the whole image. It avoids the risk of losing

some information by using only specific detector and also avoids the detection of interest points. In terms of descriptors, SIFT and SPIN descriptor for FCP and only SIFT descriptor for SCP.

2. Build the codebook(s) for different type of descriptors. We apply k -means clustering on all descriptors of every type and generate the codebook(s). The sizes of all codebooks are fixed as 500, which is also the codebook size of all methods used for comparison. Notice that hierarchical k -means clustering can be used to generate codebook and histogram of multi-resolution.
3. Assign local regions to visual phrase by and form the histogram of visual phrase frequency for every image. For every local region, it is assigned to a visual phrase by assigning every individual descriptor to some visual word of the same type separately and obtaining the K -tuple of visual phrase.

For online query processing, the histogram of visual phrase frequency is calculated in a similar way for the query image. It is then used to calculate the similarity between the query image and every image in the database. The similarity can be calculated using any distance for two histograms, e.g. $L2$ distance, χ^2 distance as well as Earth Mover's Distance(EMD), etc. In this chapter, we use the simplest intersection distance τ to measure the similarity between two histograms H_I and H_J of visual phrase frequency.

$$\tau(H_I, H_J) = \sum_{v=1}^V \min(H_v(I), H_v(J)) \quad (3.7)$$

where $H_v(\cdot)$ represents the v^{th} bin of the histogram. Although simple, the intersection distance can handle partial matching with cluttered background. Notice that Pyramid Matching Kernel(PMK) [25] can be used as distance measure instead of simple intersection for multi-resolution histogram from hierarchical codebook(s). In this chapter, we only use the simple intersection distance to show the effectiveness of the proposed coherent phrase model.

3.3.1 Complexity Analysis

Compared to the retrieval frameworks [19, 81] using the standard “Bag-of-Word” representation, the proposed method preserves the efficiency, especially for online process of queries. For the process to setup database (which is only need to do once), an additional computation is required in step 1 and 3 because we need to extract and assign multiple instead of one descriptors for every local region. Although we explore the space of visual phrase, which is a combination of K visual words, there is no need to build the codebook of visual phrase. Because we *separately* assign individual descriptor to visual word, only the codebook(s) of visual word are built, whose size is much smaller than the actual number of all possible visual phrase. For example, we only build two codebooks with the same size of 50 for SIFT/SPIN respectively and local regions can be mapped to $50 \times 50 = 2500$ different visual phrases of order 2. Compared to “Bag-of-Word” representation, the performance of the proposed model does not degrade heavily using smaller codebooks of visual word. It is because visual phrase combines K visual words and has higher discriminant power. This advantage can reduce the computation complexity of assigning local regions to visual phrases. When assigning individual descriptor of a local region to visual word, the number of required comparison is reduced by using small codebook. Notice that the process to setup database can be completed before the query is submitted and its complexity does not affect the efficiency of online process of query.

For online query processing, the proposed image representation using visual phrase preserves the sparsity of “Bag-of-Word” representation. Suppose there are M interest points in an image, the maximum number of nonzero bins in its histogram of visual phrase frequency is M , which is the same as that in “Bag-of-Word” representation. Because of this sparsity, the computation of the similarity between two images is very efficient. Notice that graph-based method like [2] requires to explicitly find the cor-

Figure 3.4: (a) Top- k IND retrieval performance of FCP and relevant methods on the first dataset. (b) Four labeled IND pairs (first 2 columns) from the first dataset and the most similar image to the images in the first column returned by the proposed method.

respondence between two images and hence has a higher computational complexity. For the methods that utilizes the relationship between n local regions, the maximum nonzero bins of the histogram in their methods are increased to M^n . For example, [21] that used the rank information about the proximity between two regions has the maximum M^2 of nonzero bins in the histogram.

3.3.2 Experiment Evaluation

In this section, we evaluate the proposed coherent phrase model of FCP and SCP, on two IND datasets which are extracted from the keyframes of TRECVID corpus [78]. The first dataset is provided by [1], which are the keyframes extracted from TRECVID 2003 corpus. The second dataset contains the keyframes extracted from TRECVID 2005 & 2006 corpus. All IND pairs as well as non-duplicate images can be viewed and downloaded from the supplementary material. Each dataset includes 150 IND pairs (300 images) and 300 non-duplicate images. The size of all the images are

Figure 3.5: (a) Top- k IND retrieval performance of SCP and relevant methods on the first dataset. (b) Four labeled IND pairs (first 2 columns) from the second dataset where SCP succeeds to return them as top-1 images but SPM fails. The top-1 images returned by SPM are shown in the third column.

352×240 . To compare with [2], we adopt the same experiment setup for IND retrieval. All IND pairs (300 images) are used as queries to evaluate performance. For each query, we calculate the similarity between the query image and all 599 images using the intersection between their histograms of visual phrase frequency. A ranked list of 599 images is then produced according to their similarity to the query. To assess the retrieval performance and compare with other methods, we estimate the probability of the successful top- k retrieval $P(k)$ as follows

$$P(k) = \frac{Q_c}{Q} \quad (3.8)$$

where Q_c is the number of queries that rank their INDs within the top- k position, and Q is the total number of queries ($Q = 300$).

3.3.3 IND Retrieval using FCP

For FCP, we use SIFT and SPIN descriptors to generate FCP of order 2. The codebook of this FCP is a 2-dimensional array, where there are 500 visual words in each

Table 3.1: The performances of top-1 IND retrieval on two dataset for different methods

Methods	FCP Model	SCP Model	SIFT BoW	SPIN BoW	SIFT+ SPIN Fusion	LIP-IS +OOS [2]	SPM [24]
Dataset 1	0.833	0.813	0.740	0.697	0.756	0.790	0.787
Dataset 2	0.960	0.960	0.920	0.843	0.937	/	0.940

dimension obtained from k-means clustering on each types of descriptors. It means that we generate 500 visual words to form the codebook of SIFT and generate 500 visual words to form the codebook of SPIN.

The proposed FCP utilizes the information of different features by enforcing the feature coherency across multiple types of descriptors for every local region. We compare it with several methods: (i) the state-of-the-art algorithm [2] which used one-to-one graph matching; (ii) the standard "Bag-of-Word" model using individual SIFT/SPIN descriptor, denoted as SIFT BoW and SPIN BoW. To prove the advantage of incorporating feature coherency using FCP, we also compare FCP with the fusion method used in [82] to combine the two "Bag-of-Word" models using SIFT and SPIN. The fusion is completed as

$$S_f(I, J) = \sum_{l=1}^K \frac{\alpha_l}{1 + \exp(-S_l(I, J))} \tag{3.9}$$

where $S_f(I, J)$ denotes the similarity between image I and J after fusion and $S_l(I, J)$ denotes the original similarity calculated in each "Bag-of-Word" model using individual descriptor e.g. SIFT/SPIN. In our experiment, the parameters are set as $K = 2$ and $\alpha_1 = \alpha_2 = 0.5$. Figure 3.4 (a) shows the results of these methods for top- k IND retrieval on the first dataset where k changes from 1 to 20. We can see that SIFT descriptor achieves better performance than SPIN descriptor when using "Bag-

of-Word” model. Although the fusion of two ”Bag-of-Word” model can improve the overall performance, the improvement is insignificant and the performance after fusion is still lower than [2]. However, the proposed FCP using SIFT and SPIN achieves a significant improvement (8% for top-1) over the fusion and outperforms [2] by (4.3% for top-1). The similar performance is also achieved on the second dataset and we show the result in the supplementary material. One problem about the first dataset is that about 5% IND pairs in the provided ground-truth are somehow confusing according to our observation. Figure 3.4 (b) shows 4 examples where the images returned by the proposed method are different from the provided ground truth. However, the returned images are actually the correct INDs from our point of view. The reported performance of the proposed method is strictly based on the provided ground truth such that it is lower than actual performance.

3.3.4 IND Retrieval using SCP

For SCP, we set $C = 2$ and use the original description scale s_0 associated with every local region and $s_1 = 2s_0$ to extract SCP of order 2 using SIFT descriptor. The codebook of this SCP is 2-dimensional array, where each dimension is a codebook of 500 visual words of SIFT descriptor.

The proposed SCP utilizes spatial information by enforcing the spatial coherency across the spatial neighborhoods of different sizes of every local region. We compare it with two methods: (i) the standard ”Bag-of-Word” model using SIFT descriptor, denoted as SIFT BoW; (ii) the Spatial Pyramid Matching (SPM) method [24] which also utilizes spatial information. Notice we do not compare SCP with the fusion method because SCP only use one type of descriptor e.g. SIFT. Figure 3.5 (a) shows the results of these methods for top- k IND retrieval on the second dataset where k changes from

Table 3.2: Average CPU time for processing a query

	LIP-IS+OOS [2]	SPM [24]	FCP/SCP
Time(s)	77.87	14.53	9.620

1 to 20. Although SPM outperforms the standard “Bag-of-Word” model using SIFT descriptor, the proposed SCP achieves even higher performance. It is because that the mechanisms of using spatial information of SCP and SPM are different. SPM constraints the matching in the corresponding spatial grids, which is not invariant to object translation/rotation and the change of viewpoint/scale. Figure 3.5 (b) shows 4 IND pairs where there exists rotation and large changes of scale/viewpoint. SCP outperforms SPM in these cases because it is fully invariant to these changes while SPM is not. A similar results is also obtained on the second dataset and is included in the supplementary material.

We also summarize the top-1 IND retrieval performances of all relevant methods on both two datasets in Table 3.1. The results are consistent across two datasets. The performance of the fusion method that combining two “Bag-of-Word” models using SIFT and SPIN is only comparable to the best performance of individual model. The LIP-IS+OOS method [2] and SPM [24] are two state-of-the-art methods which achieve better performance than the fusion method. But the proposed FCP and SCP are both significantly outperform these methods on both two datasets.

3.3.5 Time Complexity Comparison

According to the analysis of Section 3.3.1, the coherent phrase model has a low computational complexity for calculating the similarity between two images. We evaluate the average CPU times for processing a query which includes calculating similarities be-

tween the query and other 599 images for different methods. For SPM and FCP/SCP, we estimate the average time of calculating similarities using our own matlab implementation without optimization. For LIP-IS+OOS method in [2], we use the C++ implementation provided by the authors to estimate the average time. From the average times reported in Table 3.2, it is empirically proved that the coherent phrase model is not only more accurate but also more efficient than other methods. Notice that the LIP-IS+OOS method in [2] is most inefficient because it requires to find the exact correspondence for every local region when calculating the similarity between two images. The average time of SPM is longer than FCP/SCP using the codebook of same size. It is because SPM need to compare multiple pair of histograms of different sub-region in different levels while FCP/SCP only compare one pair of histograms of the whole image.

3.3.6 Discussion

Besides our FCP model, there are a few independent studies, which are similar to ours. Yuan [79] proposed a data-mining algorithm to discover the collocation patterns of visual words. Later, Zheng et. al. [83] proposed a delta visual phrase to integrate the spatial inter-relation among visual words, which is mined not only from co-occurrence information, but also the local proximity information of visual words. The above methods also increase the discriminative power of features. However, such methods are data-dependent, which requires sufficient training data and not suitable for web image retrieval. In [84], Bundled-feature groups features in local MSER [4] regions into a local group to increase the discriminative power of local features. The matching score of bundled feature sets are used to weight the visual word vote for image similarity. Since false feature matches between bundles still exist, the bundle weight will be degraded by such false matches. For our proposed coherent phrase model, either FCP or SCP,

different attributes are considered for a single point. The discriminative power is enhanced by exploring the co-occurrent information of the same point. In FCP, multiple features have to be extracted, while in SCP only one kind of feature is extracted.

3.4 Conclusion

In this chapter, we propose a coherent phrase model for image near-duplicate retrieval. Different from the standard BoW representation, this model represents every local region using multiple descriptors and enforces the coherency across multiple descriptors for every local region. Two types of visual phrase (feature coherent phrase and spatial coherent phrase) are designed to represent feature and spatial coherency. Both coherency are utilized without increasing the computational complexity. This model, although simple, improves the matching accuracy by reducing the number of false matches and preserves the matching efficiency because of the sparsity of the representation. Instead of mining the association of multiple regions which increases the complexity, the quantization error of BoW is reduced by the coherency in the visual phrase of every single region in our method. Both effectiveness and efficiency of the proposed method have been proved on multiple benchmarks built from TRECVID corpus for IND retrieval. The proposed technique can be generally applied on different variants of the BoW model.

Chapter 4

Exploiting Local Dependencies with Spatial-Scale Space (S-Cube)

4.1 Motivation

This chapter presents an efficient and effective solution for retrieving Image Near-Duplicate. Different from previous methods, we analyze the local dependencies among the descriptors in the Spatial-Scale Space (S-Cube). Such local dependencies in spatial-scale space(S-Cube) encodes not only visual appearance but also the spatial and scale co-occurrence of them. The local dependencies are exploited over the cube-space of neighboring spatial locations and multiple adjacent scales to form the new image representation, which is invariant to spatial transformation and scale change. To speed up the retrieval process, the SuperNodes are built to incorporate the neighbor information. We evaluate our proposed Spatial-Scale(S-Cube) method for IND retrieval using two existing benchmarks as well as a new dataset extracted from the keyframes of TRECVID corpus. Compared to the state-of-the-art results, our proposed local dependencies in S-Cube plus SuperNodes approach has shown a high accuracy for IND

retrieval, as well as a significant time reduction.

We propose a new model that explores local features in spatial-scale space and then form a global representation for each image. Local features make our model robust to illumination variations, viewpoint and scale changes, as well as geometric transformations, while the final global representation allows an efficient online retrieval. To be exact, see Fig. 4.1, we represent every image by calculating descriptors for each location from multiple scales. After assigning every descriptor to a specific label, we analyze the dependencies among different labels in a spatial-scale space. Such dependencies are assumed to be local to reduce the effect of noise as well as to reduce the computational complexity. We then integrate the local dependencies of labels over all spatial locations with multiple scales to form the image representation. Such representation of local dependencies encodes not only visual appearance but also the spatial and scale co-occurrences of them, which is also invariant to spatial transformation and scale changes. Finally, the SuperNodes are built to incorporate the neighbor information to speed up the retrieval process. We experimented the proposed method for IND retrieval on three datasets from the TRECVID corpus and the results show comparable performance in both accuracy and speed when compared to the state-of-the-art methods.

The rest of this chapter is organized as follows. Section 4.1 gives the detail description of the our proposed S-Cube approach. The experiment results on current near-duplicate datasets from the keyframes of TRECVID are reported and compared with other methods in Section 4.3. Finally, we conclude the contributions of this paper in Section 4.4.

Figure 4.1: The framework of our proposed system.

4.2 Spatial-Scale Space (S-Cube)

4.2.1 Multi-Scale Space feature extraction

Compared to global features, such as color histogram, color moment, local features [7] have proven to be very successful in image representation, which are not only invariant to scale and rotation changes, but also robust to changes in illumination, noise, and minor changes in viewpoint. Instead of using interest points detectors [42, 5, 85, 2], which are often suboptimal choices for certain regions, we adopt a dense sampled regular grid to obtain sufficient statistics for each image, for both blob-like and edge-like features. Also, we incorporate the scale factor when extracting features to increase the range of detectable object scale changes in Near-Duplicate images. Specifically, we use highly overlapping patches, with a scalar sampling rate of 1.5, starting from 8×8 patches up to patches of size 27×27 . For each scale, the spatial sampling step is 4

pixels.

Each of these patches is then described by a descriptor. In our experiment, we adopt the SIFT descriptor [5], which compute distributions over gradient orientations for different subpatches and is invariant and robust across a substantial range of affine distortion, viewpoint changes, noises, and different illumination. Of course, other similar invariant local descriptors, such as SPIN [9] and SURF [8] can also achieve comparable results. Meanwhile, we keep record of spatial and scale information of every feature in order to explore local dependencies.

It is very complex and time-consuming to use the descriptors directly because of their high dimension. Varying in cardinality and lacking meaningful ordering of descriptors contribute to the difficulty of finding an acceptable model to represent the whole image. To address the problems, we obtain a fixed-number label set $\mathcal{L}$ by using clustering techniques. Descriptors are assigned to their respective clusters (labels), which is the same process as standard “Bag-of-Words” method [19].

4.2.2 Spatial-Scale Space - exploring Local Dependencies

In fact, local dependencies is a common and important property for images. There is not only spatial coherency, but also scale coherency between local patches. See Fig. 4.2 for example, the patch containing the left eye often comes with the patch containing the right eye, so do the nose and mouth. In the scale space, one can find the patch containing eyes and mouth in a larger scale comparing to the scale of the patch containing nose. Local dependencies are of great importance to find a robust and discriminative representation of each image. Local dependencies are long used in solving problems of object recognition, such as spatially coherent matching[86], in which dependencies are modeled by Bayesian framework for better matching. In this

paper we try to explore the local dependencies in the spatial-scale space to form a high level representation, which is different from “Bag-of-Words” methods.

Figure 4.2: Examples of local dependencies in both spatial and scale space. Patches with red outline are from the left IND, while those with green outlines are from the right IND. (a),(b),(c) are showing horizon, vertical, and scale invariant co-occurent neighborhood relationship between the INDs, respectively.

The spatial-scale feature space is defined as

$$\mathcal{S} = \{(i, j, \sigma) \mid 1 \leq i \leq W, 1 \leq j \leq H, \sigma > 0\} \quad (4.1)$$

where W and H are the width and height of the image respectively. σ is the scale factor. The elements of the spatial-scale space correspond to the *sites* or *configurations* of each dense sampled feature. After every feature has been assigned to a label, its corresponding site in $\mathcal{S}$ will be mapped to $\mathcal{L}$ by a function f :

$$f: \mathcal{S} \rightarrow \mathcal{L} \quad (4.2)$$

where function f is to find the most similar label to the feature of current site.

Co-occurrence information of features can be explored by a neighborhood system. A neighborhood system for $\mathcal{S}$ is defined as

$$\mathcal{N} = \{\mathcal{N}_i \mid \forall i \in \mathcal{S}\} \quad (4.3)$$

where $\mathcal{N}_i$ denotes the set of sites neighboring to i . $(\mathcal{S}, \mathcal{N}) \triangleq G$ constitutes a graph, where $\mathcal{S}$ are nodes and $\mathcal{N}$ indicates links between the nodes according to their neighboring relationships. Let $\mathcal{V}_i$ denote the set of nodes excluding the node i and its neighbors $\mathcal{N}_i$ in the set of $\mathcal{S}$ for the graph G . We have the following *conditional independence* statements:

$$\{\mathcal{S}_i \perp \mathcal{V}_i \mid \mathcal{S}_{\mathcal{N}_i}\} \quad (4.4)$$

That is, given its neighbors of one node, it is independent of all the other nodes in the graph.

With the assumption of absolute independence of features for “Bag-of-Words” method, neighborhood relationship is totally neglected, which results in the loss of available matching regions. In this paper, we consider local dependencies of features. That is, not only the feature itself but also inter-relations between features in the neighborhood system are taken into consideration. Specifically, we define a second order neighborhood system in the spatial-scale feature space,

$$\mathcal{N}(i, j, \sigma_k) = \{(l, m, \sigma_n) \mid (l - i)^2 + (m - j)^2 + (n - k)^2 < 2\} \quad (4.5)$$

as shown in Fig. 4.3 (a). One node has 8 neighbors in the same scale and 9 neighbors each in both upper and lower scales. Considering a site with multi-scale changes can make the neighborhood system robustly invariant to scale changes.

We define a pair of neighboring sites i and j as the clique:

$$\mathcal{C}_{ij} = (i, j), j \in \mathcal{N}_i \quad (4.6)$$

Figure 4.3: (a) A second order neighborhood system in spatial-scale space. The dark blue node is the center and it has 8 neighbors in the same scale and 9 neighbors each in both upper and lower scales respectively. (b) An example of graph representation which integrates local dependencies in the spatial-scale space. The size of a node encodes the frequency of the label associated. And the edge $(v_i, v_j) \in E$ indicates the co-occurrence frequency between the corresponding label pair.

Labels $l_m = f(i)$ and $l_n = f(j)$ are associated with sites i and j by Eq.(4.2), and then C_{ij} will be mapped into $q_{l_m l_n}$, which shows there a transition relation from label l_m to l_n in the neighborhood system.

4.2.3 Measure for integrating Local Dependencies

In general, we explore two properties in the spatial-scale space: the frequency of a label occurring and pairwise cliques between different labels. Both are informative and discriminative. If we take these labels as different nodes, We can integrate both properties into a graph $\mathcal{G} \triangleq (V, E)$. See Fig. 4.3 (b) for example. $V = \{v_1, \dots, v_K\}$ corresponds to the K nodes, the size of which denotes the frequency of the label associated. The edge $(v_i, v_j) \in E$ denotes the dependence between node v_i and node v_j , which integrates the local co-occurrence frequency between the two labels over all

possible locations in spatial-scale space. In order to compute the node, we adopt the Delta function

$$\delta(i, l) = \begin{cases} 1 & \text{if } i = l \\ 0 & \text{else} \end{cases} \quad (4.7)$$

So the frequency of node k is,

$$p_k = \frac{1}{N} \sum_{i=1}^N \delta(f(i), k) \quad (1 \leq k \leq K) \quad (4.8)$$

where $\sum_{k=1}^K p_k = 1$ and N is the total number of sites. And the transition probability from node k to node l is

$$p_{kl} = \frac{\sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \delta(f(i), k) \delta(f(j), l)}{\sum_{i=1}^N \delta(f(i), k)} \quad (4.9)$$

The term $\sum_{i=1}^N \delta(f(i), k)$ is the normalization factor to satisfy $\sum_{l=1}^K p_{kl} = 1$.

Under the graph $\mathcal{G}$, the similarity of two images I_1 and I_2 is naturally defined as

$$Sim(I_1, I_2) = \alpha \sum_{k=1}^K \mathcal{I}(k) + (1 - \alpha) \sum_{(v_i, v_j) \in E} (\mathcal{D}_{ij}) \quad (4.10)$$

where $\mathcal{I}(k)$ is the function measuring the similarity of each node v_k , while the function $\mathcal{D}_{ij}$ measures the similarity of the co-occurrence relationship of two nodes v_i and v_j in different images. α and $(1 - \alpha)$ are the weighting coefficients of the two parts. The similarity function of $\mathcal{I}$ and $\mathcal{D}$ can be calculated using any distance, e.g. $L2$ distance, χ^2 distance as well as EMD distance, etc. Note that the standard “Bags-of-words” method is a special case of our proposed method when we set $\alpha = 1$ in Eq. (4.10).

4.2.4 Speed-Up the retrieval with SuperNodes

The graph $\mathcal{G}$ contains N nodes and N^2 edges, computing similarity of two graphs would requires $N^2 + N$ comparisons. In this section, we try to explore the details of the graph in order to get a compact representation for better computing. That is, we

want to get rid of links either sensitive or non-informative. First, we remove edges with small transition probability, which means there are little or no frequent co-occurrence between the nodes linked by the edge. These edges are easy to incorporate noise. Second, from our observation, we find that the links from one node to itself is very common in image, which result in such links less discriminative and non-informative. After pruning these links, the discriminative power will be enhanced.

Obviously, searching speed is a key factor to judge a IND retrieval algorithm. In order to speed up our S-Cube, we propose an SuperNode here. For a node in graph, we do not compare all the edges one by one, which implicit co-occurrence information to other nodes. Instead, we integrate all these links to one, as a complementary part of the node. This is similar to [87], in which the new node contains not only information of itself but its neighbors. The enhancement we use here is an integration of stable and discriminative co-occurrence neighbors. That is,

$$E_k = \sum_i (p_i * p_{ik}) \quad (i \in [1, \dots, K], i \neq k \text{ and } p_{ik} > T) \quad (4.11)$$

$p_{ik} > T$ is to exclude the neighbors, which is less co-occurrence with current node. T is the minimum confidence of co-occurrence. With the SuperNode, now the comparisons reduce to $N + N$. In the theory of Markov chain, $p_i * p_{ik}$ is the transition rate from node i to node k , which describe dynamics effects of co-occurrence. E_k can be seen as the total transition rates from other nodes to node k , which characterizes the inter-nodes co-occurrence information.

Correspondingly, $\mathcal{D}$ from Eq.(4.10) will measure the similarity of E_k instead of p_{ij} . In this paper, we use the χ^2 -distance to measure the similarity for both $\mathcal{I}$ and $\mathcal{D}$, which has been proven to be a very effective similarity measurement of histogram-based representation[88]. Given two histograms $H_1 = (x_1, \dots, x_n)$ and $H_2 = (y_1, \dots, y_n)$, the χ^2 -distance is defined as

Figure 4.4: Illustration of SuperNode.

$$D(H_1, H_2) = \frac{1}{2} \sum_{i=1}^n \frac{(x_i - y_i)^2}{(x_i + y_i)} \quad (4.12)$$

To combine $\mathcal{I}$ and $\mathcal{D}$, the logistic fusion function as follows is used to reach convergence:

$$S(i, j) = \frac{\beta}{1 + \exp(-\mathcal{I})} + \frac{(1 - \beta)}{1 + \exp(-E)} \quad (4.13)$$

Our proposed SuperNodes speed up the retrieval by representing the image in a brief way, which is independent of the indexing method used for matching in a large-scale database, such as locality sensitive hashing(LSH), inverted file. In fact, the index approach can be easily used for our proposed SuperNode, which will definitely further speed up the retrieval process.

4.3 Experiments

We follow a framework similar to [19, 81] for IND retrieval. It consists of two parts: the process for database setup and the process for handling input query. First we build the graph representation of each image referring to Sec 4.1. Then for online query

Figure 4.5: Evaluation scale space and local dependencies on the Columbia dataset and CityU dataset. The green curve is the result by using only one scale features and the blue one is the result with multiple scale features. The red one shows the result considering both multiple scale features and local neighborhood dependencies. Note: the accuracy in (b) decreases due to multiple NDKs of one keyframe.

processing, the SuperNode strategy is adopted for the query image. It is then used to compute the similarity between the query image and every image in the database.

In this section, we evaluate our proposed S-Cube method on three IND datasets which are extracted from the keyframes of TRECVID corpus [78]. The Columbia dataset is provided by [1], which are the keyframes extracted from TRECVID 2003 corpus. The NTU dataset used in [89] contains the keyframes extracted from TRECVID 2005 & 2006 corpus. Each dataset includes 150 IND pairs (300 images) and 300 non-duplicate images. The size of all the images are 352×264 . The third dataset CityU dataset is a much larger dataset from TRECVID 2004 corpus, with 7006 keyframes including 3388 near-duplicate image pairs. In this dataset, there exist more than one near-duplicate images for some keyframes. To compare with other state-of-art method, we adopt the same experiment setup for IND retrieval. All IND pairs are used as queries to evaluate performance. For each query, we calculate the similarity between the query

image and all the other images. A ranked list is then produced according to their similarity to the query. To access the retrieval performance our proposed method and compare with other state-of-art methods, we estimate the probability of the successful top- k retrieval $P(k)$ as follows

$$P(k) = \frac{Q_c}{Q} \quad (4.14)$$

where Q_c is the number of queries that rank their INDs within the top- k position, and Q is the total number of queries.

4.3.1 Experiment Settings

Specifically, we use the overlapping dense sampling patches, with a scalar sampling rate of 1.5, starting from 8×8 patches up to patches of size 27×27 . For one scale, the spatial sampling step is 4 pixels. For S-Cube, we adopt SIFT descriptors to generate graph representation based on neighborhood system. 500 labels are obtained from k-means clustering algorithm on the descriptors. The proposed S-Cube utilizes the information of nodes frequency as well as nodes co-occurrence, by exploring spatial-scale space neighborhood system.

4.3.2 Scale Effects and Local Dependencies' Study

In this section, we study if scale space and local dependencies information help to improve the retrieval performance. Experiments are performed on the Columbia Dataset and CityU Dataset. We compare the performance of three strategies. The first one is only in spatial space, that is, only one scale. The second is in spatial-scale space, but with nodes only. The last one is also in in spatial-scale space and both nodes and edges are considered.

The results are shown in Fig. 4.5. We can see that the performance is significantly

improved when we exploit the scale space and it is further improved when we consider local dependencies in spatial-scale space.

4.3.3 Comparison of IND retrieval results

We compare our method with other state-of-the-art methods: (i) the NIM [85] (ii) the S^3VM [85], (iii) the LIP-IS+OOS [2], (iv) the OOS-SIFT approach [90] (v) the Visual Keywords(VK) methods [90] and (vi) color moment [91] or color histogram [1] based retrieval . Figure 4.6 shows the results of these methods for top- k IND retrieval on the Columbia dataset and cityU dataset where k changes from 1 to 30. From the figure, we can see that our proposed S-Cube is comparable to NIM and outperforms the other methods greatly in terms of accuracy. In addition, our S-Cube outperforms NIM in terms of efficiency. Note that S-Cube uses only one local descriptor, while NIM adopts not only local descriptor but also four kinds of global features, that is, Grid Color Moment, Local Binary Pattern, Gabor Wavelets Texture and Edge.

Figure 4.6: Top- k IND retrieval performance of S-Cube and relevant methods on the Columbia dataset and CityU dataset. Note: the accuracy in (b) decreases due to multiple NDKs of one keyframe.

Table 4.1: Top-1 retrieval performance comparison of S-Cube, SAPM, SPM and TPM on both Columbia Dataset and NTU Dataset. For SPM, TPM and SAPM, the first number is the performance using only one level, and the second is the highest results using multilevel with different fusion methods.

	Columbia Dataset	NTU Dataset
SPM(%)	76.3/78.0	82/85.3
TPM(%)	74.7/75.7	87/84.0
SAPM(%)	79.0/80.7	87.7/88.3
S-Cube(%)	85.0	92.0

We also compare our S-Cube approach with Spatial Pyramid Matching (SPM) [24], Temporal Pyramid Matching (TPM) and Spatially Aligned Pyramid Matching (SAPM) [89], which are related methods to ours and also incorporate spatial information. Table 4.1 shows the top-1 retrieval performance on the Columbia Dataset and NTU dataset. In both dataset, our results significantly outperform the other three approaches, which shows that our strategy to use local dependencies in both spatial and scale space provides more discriminant information for determining IND.

Fig. 4.7 shows some challenging results successfully returned by our proposed approach. These near-duplicate keyframes undergo various changes, such as illumination difference, scale variance, different editing, view point changes, motion, occlusion and so on. Because we incorporate the scale information, our proposed S-Cube is invariant to scale changes within a relative large range. Meanwhile, we consider only the local dependencies and no global constraints so that our proposed approach is more robust to viewpoint changes, motion, occlusion and so on.

We also show some challenging queries which failed in our proposed approach in Fig. 4.8. Generally speaking, there are two typical difficult problems. When one image zooms in too much or suffers from heavy editing, the near-duplicate part is only a small part of the image, which make it difficult. Another problem is about the accuracy of the

groundtruth. Some of the results from our method do not appear in the groundtruth, but visually the suitable IND candidates of the query image.

Figure 4.7: Some challenging results successfully returned by our proposed approach. These near-duplicate keyframes suffers from various changes, such as illumination difference, scale variance, different editing, view point changes, motion, occlusion and so on.

4.3.4 Evaluation of online computation cost

In this section, we examine the efficiency performance of our proposed S-Cube. Feature extraction, local dependency graph and superNodes are completed offline. The online retrieval time for all queries of Columbia dataset and CityU dataset are summarized in Table 4.2 and Table 4.3, respectively. From the results presented in the respective tables, it is observed that our proposed S-Cube is much faster than NIM, OOS-SIFT, S^3VM and VK. The online retrieval time of SAPM is even much longer than OOS-SIFT. For IND retrieval, both efficiency and speed are important considerations, it is therefore promising that our S-Cube has a fast retrieval speed while preserving a high

Figure 4.8: Some failed cases returned by our proposed approach. The first row are queries(Q), and the second and third row are groundtruth near-duplicate(G) and keyframes returned by our method(R), respectively.

Table 4.2: Comparison of overall time cost of 300 queries on the Columbia dataset

Our S-Cube	NIM	S^3VM	OOS	VK
2sec	15.8min	3sec	6.5hour	1.5min

accuracy.

4.3.5 IND guided video retrieval - Initial results

In this section, we present some initial results for video retrieval and video event detection using the matching of multiple INDs. Given multiple video sources, users

Table 4.3: Comparison of overall time cost of 1953 queries on the CityU dataset

Our S-Cube	NIM	S^3VM	OOS	VK
4.5min	103.5min	8.1min	N/A	N/A

might be interested to search for current/interesting events reported by different sources and with different viewpoints. For example, the news focus of the current period across multiple broadcast sources is likely to be the most interesting event. All the INDs in the CityU dataset are extracted from 52 groups within the CNN and ABC broadcast videos. There are 27 video shots from CNN and 25 video shots from ABC. The INDs which were identified by using the proposed method are utilized to retrieve the common events from multiple video sources. We consider not only individual IND but also the temporal relationship between neighboring INDs to discover common event. Fig. 4.9 shows the event retrieval results from multiple video shots with the time label from 1998.03.02 to 1998.03.31. As observed, different events are retrieved by grouping different number of INDs and the corresponding events are across multiple video sequences. Although the temporal occurrences of different events are dissimilar in different videos due to editing operation, we can still detect and retrieve those common events.

Figure 4.9: IND(s) guided video event detection (of the same event) among different videos. Video (a) is news clip from CNN. Video (b) & (c) are news clips from ABC but different date/time stamp.

4.4 Conclusion

In this paper, we propose to analyze the local dependencies in our spatial-scale space (S-Cube representation), which incorporate not only the appearance feature information, but also the spatial and scale co-occurrence information. Through the integration of local dependencies over neighboring spatial locations and multiple scales, the proposed S-Cube method is invariant to spatial transformation and a wider range of scale change. Experiments on three IND datasets have shown that S-Cube is a promising approach for IND retrieval. In addition to improving retrieval accuracy, a faster retrieval speed has shown that our proposed S-Cube has great potential in Near-Duplicate image retrieval and IND assisted video retrieval.

Chapter 5

Stratification-based Keyframe Cliques for Effective and Efficient Video Representation

5.1 Introduction

As the exponential increase of web videos, it is time-consuming to get a query result from the tremendous data. An effective and efficient video management system is in urgent need. To increase the efficiency of video retrieval and storage, the most widely used methods are indexing schemes, such as Locality Sensitive Hashing (LSH). However, it is more essential to represent the video itself compactly. In this chapter, we propose a strategy to generate Stratification-based Keyframe Cliques (SKCs) for video description, which are more compact and informative than frames or keyframes. The new representations are scalable for different retrieval tasks due to the ranking of SKCs. To further accelerate the retrieval speed, only top SKCs will be used, meanwhile, the searching results are still satisfactory. Experiments are conducted on TRECVID

dataset as well as web video dataset. Results show that our proposed SKCs are more succinct and informative for video retrieval and management.

5.2 Motivation

With the growth of applications in multimedia technology, video data has become one of the fundamental resources in modern databases. On the web, there are an increasing number of websites allowing users to broadcast videos themselves. As an example, YouTube¹ is a video sharing website where users can upload, browse and share video clips. The wide range of topics covered by YouTube has turned video sharing into one of the most important parts of Internet culture. However, due to ease of uploading users' videos, a new set of problems for managing videos has surfaced. Many videos are posted without enough tags for effective management, some are even posted without tags or any form of descriptions. One of the new problems that is deemed as significance is the substantial increase in videos with very similar contents, such as exact copy, temporal edited, spatially resized and so on. The near-duplicate videos may be uploaded many times from many different users. What's more, many unauthorized clips from television shows, films and music videos are uploaded. So the problem of efficient identification of near duplicate videos (see Fig. 5.1 for example) on the web is an important issue for video management.

There has been a great deal of work in the area of video management. The direct way is to compare frames between videos. However, using every frame for video understanding is quite time consuming and unnecessary. Removal of temporal redundancies in video has been shown efficient in video compression techniques. Therefore, many researchers start from a good segmentation of video shots and select certain

¹Youtube available: <http://www.youtube.com/>

Figure 5.1: Examples of Near-Duplicate Videos from TRECVID and Webvideo dataset.

frames of the shots as keyframes. For keyframe comparison, there are usually two strategies, one is to employ global features extracted from each keyframe, such as color histogram [92, 93, 94] or color moment [1, 95]. Although these methods are efficient, they may not be accurate enough as they fail to address lighting changes, viewpoint changes, and occlusions. Meanwhile, local features [5, 7] have been shown to be robust to lighting, viewpoint and scale changes. However, methods based on local feature matching [23, 1, 2, 85] suffer from heavy computational cost. In the research area of object classification and detection, “Bag-of-Words” (BOW) [19] model is widely used. Under this model, “visual words” are assigned to local regions. Images are treated as documents and represented by a histogram of words frequency. That is, BOW explores local features and forms a global representation.

Different from former researchers [30, 31, 32], who mainly focus on better keyframe comparisons, we try to find a new compact and informative representation of videos. Though keyframes from shot boundary detection for video representations have been

Figure 5.2: The framework of our proposed system.

shown capable, there still exist many similar keyframes in a single video clip, see Fig. 5.1 for example. Such similar keyframes can be seen as Near-Duplicate Keyframes (NDKs). Near-duplicate keyframes are common in many videos due to kinds of reasons, such as inefficient shot detection algorithm, periodical action, camera moving to-and-fro, montage, etc. The NDKs cost extra storage and decrease the retrieval speed. Also, they are challenging for keyframe alignment methods [32]. Another weakness of keyframes from shot boundary detection is that all the keyframes are taken as the same importance to the video and cannot be further compressed.

Our objective in this chapter is to propose a new representation for video clips, which is more compact and efficient than traditional keyframes. For each video clip, similar keyframes are summarized into one clique by analyzing their visual contents. And to measure the importance of the clique, the number of keyframes in the clique is adopted as the weight. With the weighted cliques, we represent each video clip (see Fig 5.2 for illustration). We also propose a new video distance based on our SKCs to make them scalable for different retrieval tasks, with which the top SKCs can be used

to measure the similarity of different videos. Generally speaking, there are two merits for our proposed SKCs:

1. Our SKCs are more compact. Each SKC is a summarization of all the near-duplicate frames/keyframes in one video, while traditional keyframes from shot boundary detection are only for consecutive frames.
2. Our SKCs are scalable. As the SKCs are weighted according to their importance to the video clip, the SKCs with higher weight contain more information. Thus, we can select different ratios of SKCs for different retrieval tasks to make a trade-off between the speed and accuracy.

This chapter is organized as follows: Some related work in video retrieval and near duplicate keyframe detection is introduced in Section 5.3. Section 5.4 proposes our novel approach of keyframe cliques to detect near-duplicate videos. Experimental results based on our proposed method in multiple video datasets is presented in Section 5.5. Finally, the conclusion and discussion are given in Section 6.6.

5.3 Related Work

Near-duplicate keyframe retrieval and detection become a hot research topic these years. Many researchers [1, 23, 2, 85] proposed fruitful approaches to improve the performance of NDK detection. Among them, Ke et al. [23] represented each image as a set of local covariant regions [5, 7], each of which is characterized by *PCA-SIFT* descriptor. Locality sensitive hashing (LSH) [96] is proposed to design an efficient index structure for fast point set matching. Later, Zhao et al. [2] extended the matching of local feature set by introducing one-to-one symmetric property for matching and LIP-IS index structure for a fast and approximate search. Recently, with the widely-

used “Bag-of-Words” methods [19, 90], near-duplicate keyframe detection achieves a better performance and is more efficient in computation time. Motivated by the successful near-duplicate detection techniques and based on the fact that near-duplicate keyframes exist in the same video, we propose a stratification-based keyframe cliques method to re-organize the keyframes for detection and removal of near duplicate web videos.

5.4 Stratification-based Keyframe Cliques

Similar to image detection/retrieval, the computational speed requirement for near-duplicate video detection/retrieval is stringent. We have visually observed that there are a significant amount of near-duplicate keyframes in each video. Such a substantial number of near-duplicate keyframes not only reduce the detection speed, but set additional constraints for keyframe matching method. In this section, we are proposing a novel stratification-based keyframe clique mining approach for the removal of near-duplicate videos.

5.4.1 Stratification-based keyframe cliques

Due to the widely existence of near-duplicate keyframes in videos, it is worthwhile detecting and grouping them. To model videos for retrieval, storage or summarization, two main approaches are relied on, that is, the segmentation-based approach and stratification-based approach [97]. The segmentation-based approach segments a video clip into sections of contiguous frames, while stratification-based approach [98, 99] models video content into chunks of contextual information. The stratification-based model views video segments as layers of strata, which may overlap, be contained in, or

even encompass other descriptions.

It is common that similar keyframes exist in one video and occur over multiple time series. For near-duplicate keyframes in a video clip, the idea to represent them as a stratum is appropriate. Such strata of near-duplicate keyframe cliques will have contextual meanings and are useful for video summarization and modeling. In our current work, we use stratification-based keyframe cliques for video representation and its applications, such as video retrieval.

5.4.2 Near Duplicate Keyframe (NDK) detection

The original BOW method for NDK detection has its limitations. One of the shortcomings is the loss of spatial information when local covariant regions are represented as a histogram of visual words. In order to explore the spatial information, we follow the Spatial Pyramid Matching (SPM) scheme proposed by Lazebnik et al.[24], which works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region. The final “spatial pyramid” representation is a simple and computationally efficient extension of an orderless bag-of-words image representation, keeping loose spatial layout of local features. See Fig. 5.3 for illustration. Different from Lazebnik et al. [24], we consider not only the local covariant features, but also global features. That is, the global color histogram is also adopted for each subregion, as well as the BOW histogram.

With spatial pyramid representation, we explore multi-resolution matching to measure the similarity of keyframes. In this chapter, we use the χ^2 -distance for each level of the spatial pyramid representation. Exactly, given two histograms $H_1 = (x_1, x_2, \dots, x_n)$ and $H_2 = (y_1, y_2, \dots, y_n)$, the χ^2 -distance is defined as

$$D(H_1, H_2) = \frac{1}{2} \sum_{i=1}^n \frac{(x_i - y_i)^2}{(x_i + y_i)} \quad (5.1)$$

Figure 5.3: Spatial pyramid representation of keyframe of different levels. (a) level $l = 0$; (b) level $l = 1$; (c) level $l = 2$.

To consolidate the distance measure for different levels, we compute the similarity of two keyframes I_1 and I_2 using the following kernel function [88]:

$$S(I_1, I_2) = \exp\left\{-\frac{1}{\beta} \sum_{l=1}^L D_l(H_1, H_2)\right\}, \quad (5.2)$$

in which β is a scaling parameter and $\beta > 0$. L is the pyramid level in total. Based on Eqn. 5.2, whether two keyframes are considered as a near-duplicate keyframe (NDK) pair can be judged by a threshold S_T , which is either given by the user or learnt with a set of near-duplicate keyframe pairs and non-near-duplicate keyframes.

5.4.3 Strata partition with quasi-clique

Based on the near-duplicate keyframe detection, each video will form a graph. Links between keyframes are generated with the threshold S_T . If $S(I_1, I_2) > S_T$, there is a link between keyframe I_1 and I_2 . Our aim here is to partition the graph into several subgraphs (cliques), in which the keyframes are NDKs. However, the issue is that the NDKs cannot always form a complete clique. Take Fig. 5.4 (i) as an example, keyframes A, B, D, E are NDK groups, while Keyframes B, C, D are also NDK groups. But A & E are not NDK pairs with C , that is, there is no link between A & C nor E &

C. Generally, this phenomena results from three factors. First, due to the montage, editing and rearrangement, some transitional keyframes link to keyframes which are not near duplicates (Fig. 5.4 (i)). Second, when the camera is moving from one side to another, or there is an moving object in the shot, the keyframes will change gradually and form a connected graph (Fig. 5.4 (ii)). However, the beginning keyframes and the ending ones are unrelated. Third, the wrong keyframe pair detection may link two irrelevant near-duplicate groups into one (Fig. 5.4 (iii)). In all, the above issues make the partition problem complicated. Do we consider the keyframes as separate complete cliques or a single quasi-clique? To answer the above question, two criteria are given:

1. On one hand, we want to maintain the discriminative power of each NDK group. That is, each group is informative and all the keyframes in it should contain a specific and similar meaning.
2. On the other hand, we want to divide one video clip into less NDK groups so that we can benefit from the brief representation for efficient detection of near-duplicate videos.

In this section, we propose a quasi-clique mining method. Different from a complete clique, in which there is a link between any vertices, in quasi-clique it is the case that most of the vertices have links with each other, but not all. In order to well explain the problem, we use $G = (V, E)$ to represent the undirected graph of a video. And the number of vertices and edges in the graph are $|N_v|$ and $|N_e|$ respectively. It is easy to know that the maximal number of edges $|N_e|_{max}$ for any graph with $|N_v|$ vertices is

$$|N_e|_{max} = \frac{|N_v|(|N_v| - 1)}{2}. \quad (5.3)$$

A quasi-clique is a graph in which the number of edges is close to the maximal number of edges. Based on N_v and $|N_e|_{max}$, we introduce the **defectiveness** of graph G as

Figure 5.4: Examples of overlap cliques due to different factors.

follows:

$$defective(G) = \frac{|N_e|_{max} - |N_e|}{|N_v|}. \tag{5.4}$$

So the quasi-clique is defined as:

Quasi-clique: Given a threshold t , if $defective(G) \leq t$, Graph G is a quasi-clique. Specifically, if $defective(G) = 0$, then graph G is a clique.

Our aim is to divide the graph G from one video clip into quasi-cliques g . To make it easy, we add another criteria: There is no overlap between quasi-cliques. Mathematically, it notes as $g_i \cap g_j = \Phi$, $(i \neq j)$. Based on the criteria above, the

following optimization problem is obtained:

$$\begin{aligned}
 & \text{minimize } M \\
 & \text{subject to } G = g_1 \cup g_2 \cup \dots \cup g_M \\
 & g_i \cap g_j = \Phi, \quad i \neq j \ \& \ i, j \in [1, 2, \dots, M] \\
 & \text{defective}(g_i) \leq t, \quad i \in [1, 2, \dots, M]
 \end{aligned} \tag{5.5}$$

in which g_i is a quasi-clique and M is the number of quasi-cliques in a video. It is NP-hard to solve the above problem. However, for efficient mining of quasi-cliques, an approximation scheme as shown in Algorithm 1 is used. The notations used throughout this algorithm are summarized in Tab. 6.1. Generally speaking, the algorithm has two steps. First, to reduce the computational complexity, the graph G of a video is divided into several connected subgraphs G_i and there are no links between any two subgraphs. Second, every subgraph is partitioned into quasi-cliques with a greedy algorithm. At the beginning, the vertex with the maximum degree is chosen as the seed. The closest vertex is selected and checked if it can be merged. The stopping criterion is that all the vertices are assigned into one of the quasi-cliques.

Table 5.1: Notations used throughout this algorithm of quasi-clique mining.

symbol	description
G	the graph from one video clip
G_i	the i -th connected subgraph of G
u, v	vertex of the graph
$\text{defective}(g)$	the defectiveness of graph/subgraph g
$\text{deg}(v)$	the degree of vertex v
$\text{Neighbor}(v)$	the neighbors of Node v
$\text{closeness}_S(v)$	the closeness of Node v to subgraph S

In Algorithm 1, $\text{deg}(v)$ is the degree of Vertex v . To be exact, it is defined as the number of links incident upon a vertex. In graph theory, closeness is a centrality measure of a vertex within a graph. Vertices that are “shallow” to other vertices (that

is, they have short geodesic distances) have higher closeness. Here the closeness is defined as follows,

$$closeness_S(v) = \frac{|Neighbor(v) \cap S|}{|Neighbor(v)|}, \quad (5.6)$$

in which $Neighbor(v)$ is the neighbors of Node v . That means, the larger proportion of neighbors for one vertex fall into S , the closer the vertex is to S .

Algorithm 1 Mining Quasi-clique Algorithm.

```

1:  $G = \{G_1, G_2, \dots, G_i, \dots, G_K\}$ 
   //  $G_i = (V_i, E_i)$  is connected subgraph of  $G$ 
   //  $G_i \cap G_j = \emptyset$  when  $i \neq j$ 
2: FOR  $k$  FROM 1 TO  $K$ 
3:    $G_k = (V_k, E_k)$ 
4:   WHILE  $V_k \neq \emptyset$ 
5:      $seed = \arg \max_{v \in V_k} deg(v)$ 
6:      $S = seed;$ 
7:      $N_s = \{v | (u, v) \in E, u \in S, v \notin S\}$ 
8:     FOR  $v \in N_s$ 
9:        $closeness_S(v) = \frac{|Neighbor(v) \cap S|}{|Neighbor(v)|}$ 
10:     $S_v = \arg \max_{v \in N_s} closeness(v, S)$ 
11:    IF  $defective(S_v + S) \leq threshold$ 
12:       $S = S_v \cup S$ 
13:       $N_s = \{v | (u, v) \in E, u \in S, v \notin S\}$ 
14:    ELSE
15:      remove  $S$  from  $V_k$ , break
16:    END FOR
17:  END WHILE
18: END FOR.
```

5.4.4 Efficient representation of SKCs

After partitioning, each video clip is represented by several quasi-cliques. Without loss of generality, we call it Stratification-based Keyframe Cliques (SKCs) in the rest of the chapter. As the keyframes in each SKC are near-duplicates, the spatial pyramid representation for keyframe can logically extend to describing SKCs. Near-duplicate

keyframes in a SKC exhibit similar spatial and multi-level layout. For efficient detection, we represent each SKC with the mean of all the NDKs in it. See Fig. 5.5 for illustration. That is, for a clique with K keyframes, $SKC = \{I_1, I_2, \dots, I_K\}$, the histogram bins at each level for a SKC are the average of the corresponding bins for all the individual keyframes at the respective level :

$$\overline{H}_l = \frac{1}{K} \sum_{i=1}^K H_{li} \quad (5.7)$$

Note that the mean of keyframes is not the sole representation for a SKC. Other strategies, such as selecting one keyframe randomly from the SKC, are also applicable.

At the same time, a weight $w = K/N_v$ is attained regarding to the number of keyframes in each SKC. For each video clip, we have $G = \{(g_1, w_1), (g_2, w_2), \dots, (g_M, w_M)\}$, in which w_k is the weight associated with the corresponding SKC and

$$\sum_{i=1}^M w_i = 1. \quad (5.8)$$

The weight w measures the importance of the SKC to the video. It is believable that a SKC with larger weight catches more information and is more stable in the video. It has higher probability to occur in the near-duplicate videos. Note M is expected to be much smaller than N_v so that SKCs will greatly speed-up the detection process.

5.4.5 Video distance based on SKCs

With compact SKC representations for each video clip, we proceed with the task of near-duplicate video detection and removal. The similarity between the query video clip and all the candidates video clips is computed. For multi-dimensional distributions, the Earth Mover's Distance (EMD) [100, 9, 88] is a widely-used approach. However, due to its computational complexity, it is inefficient for large scale video datasets. In this section, we propose an efficient method to obtain the video similarity.

Figure 5.5: Stratification-based Keyframe Cliques. (a) The keyframes of one video; (b) Near-duplicate Keyframe Cliques; (c) The compact representation of NDK Cliques.

Given two video clips with weighted SKC representations $G^i = \{(w_{i1}, g_{i1}), (w_{i2}, g_{i2}), \dots, (w_{iP}, g_{iP})\}$ and $G^j = \{(w_{j1}, g_{j1}), (w_{j2}, g_{j2}), \dots, (w_{jQ}, g_{jq})\}$, we try to define a distance to measure the similarity between them. First, we introduce a clique-to-video (CTV) distance, which is obtained by calculating the nearest distance from the clique g_{ip} in video clip i to all the cliques $g_{jq}, q \in [1, \dots, Q]$ in video clip j . Exactly,

$$D(g_{ip}, G^j) = \min_{q \in [1, \dots, Q]} D(g_{ip}, g_{jq}). \quad (5.9)$$

Consider the weight of each clique, the distance from video i to video j is,

$$D_{i \rightarrow j} = \frac{\sum_{p=1}^k w_{ip} D(g_{ip}, G^j)}{\sum_{p=1}^k w_{ip}}, \quad 1 \leq k \leq P \quad (5.10)$$

Here k is no need to be equal with P . One may prefer to using part of SKCs for efficiency consideration. From Eqn.5.10, we see the distance is not symmetrical. That is, the distance from video i to video j is not always equal to the distance from video j to video i . To make it symmetric, we define the distance between video i and video j as follows.

$$D_{ij} = \frac{1}{2}(D_{i \rightarrow j} + D_{j \rightarrow i}) = \frac{1}{2} \left(\frac{\sum_{p=1}^{k_i} w_{ip} D(g_{ip}, G^j)}{\sum_{p=1}^{k_i} w_{ip}} + \frac{\sum_{q=1}^{k_j} w_{jq} D(g_{jq}, G^i)}{\sum_{q=1}^{k_j} w_{jq}} \right). \quad (5.11)$$

If two videos are near-duplicate, most of their SKCs (especially those with higher weight) are correlative. Otherwise, their SKCs are irrelevant.

5.4.6 Complexity Analysis

Different from the traditional keyframe based retrieval, which directly compare all the keyframes between each video pair, our algorithm has two steps: First, analyze the keyframes in one video and group them into SKCs. Second, compare the SKCs between each video pair. The first step is processed offline. What we focus on is the second step. As the videos are represented with compact SKCs instead of keyframes, not only the storage space is saved, but also the retrieval process is sped up. If one SKC contains n keyframes averagely, the storage for SKCs covers only $1/n$ of that for keyframes. Correspondingly, the time cost of retrieval is only $1/n^2$ of the original. What's more, the SKCs are ranked, which make it scalable for retrieval. The retrieval speed can be further accelerated if we use only the top SKCs. Exactly, if only the first $1/k$ proportion of the SKCs are adopted, the time cost is $1/k^2$ of that using all the SKCs. Comparing with retrieval using keyframes, the total time of using the first $1/k$ proportion of the SKCs will be reduced to $\frac{1}{n^2 k^2}$.

5.5 Experiments Evaluation

5.5.1 Dataset and Keyframe Representation

To conduct comprehensive evaluations, we use two benchmark datasets for our experiment. The first dataset is created from TRECVID 2005 and TRECVID 2006 datasets. From each dataset, the first 20 videos are used to evaluate the summarization of keyframes. The videos are news videos from multiple corpus, such as CNN, NBC, CCTV, LBC, NTDTV and so on (see Fig. 5.6). The second dataset is provided by Wu [31] and crawled from YouTube, Google Video and Yahoo Video. It is used to gauge the performance of our algorithm for detection/removal of near-duplicate web-video clips. In all, there are a number of 12,790 web-video clips. The benchmark dataset is organized into 24 search queries. Details are shown in Table 5.2.

Figure 5.6: Video shots examples from TRECVID 2005 & 2006. (a)-(e) are the keyframes of 5 different video clips.

Spatial pyramid representation is adopted to handle the issue of spatial information.

Table 5.2: Video Queries Collected by Wu from YouTube, Google Video and Yahoo Video. $V\#$ and $ND\#$ mean number of videos and number of near-duplicate videos respectively.

ID	Query Name	#	ND#	%
1	The lion sleeps tonight	792	334	42%
2	Evolution of dance	483	122	25%
3	Fold shirt	436	183	42%
4	Cat Message	344	161	47%
5	OK go here it goes again	396	89	22%
6	Urban ninja	771	45	6%
7	Real life Simpsons	365	154	42%
8	Free Hugs	539	37	7%
9	Where the hell is Matt	235	23	10%
10	U2 and green day	297	52	18%
11	Little superstar	377	59	16%
12	Napoleon dynamite dance	881	146	17%
13	I will survive Jesus	416	387	93%
14	Ronaldinho ping pong	107	72	67%
15	White and Nerdy	1,771	696	39%
16	Korean karaoke	205	20	10%
17	Panic at the disco I write sins not tragedies	647	201	31%
18	Bus uncle	488	80	16%
19	Sony Bravia	566	202	36%
20	Changes Tupac	194	72	37%
21	Afternoon delight	449	54	12%
22	Numa Gary	422	32	8%
23	Shakira hips don't lie	1,322	234	18%
24	India driving	287	26	9%
	Total	12,790	3,481	27%

Both global color histograms and local covariant features are considered for keyframe representation. In the following experiments, they are linearly concatenated. For global feature representations, we use color histogram (CH). A color histogram is a representation of the distribution of colors in an image, derived by counting the number of pixels for each color ranging in the color space. To compute the color histogram, the widely used HSV color space is adopted, which comprises three components: hue, saturation and value (brightness). To be exact, an image patch is represented as a histogram, concatenated with 18 bins for hue, 3 bins for saturation, and 3 bins for value. The detection and description of local image features are well explored in computer vision and multimedia. Many effective features [42] have been proposed to date, which are invariant to image scaling and rotation, as well as robust to changes in illumination, noise, and viewpoint. Among them, SIFT feature [5] is one of the most widely used features in BOW applications. We use the extended PCA-SIFT proposed by Ke et al [6], which reduces the feature dimensions from 128-dimension to 36-dimension. For keyframe representation, we use a three-level spatial pyramid, that is $L = 2$. There are 2^{2^i} subregions for level i ($i \in [0, 1, 2]$) and each subregion is represented by both global color features and local PCA-SIFT based BOW histograms.

5.5.2 Evaluation of Keyframe Summarization

In this part, we try to evaluate the compactness of SKCs. The proposed quasi-clique mining algorithm is applied to the dataset from TRECVID2005 and TRECVID2006 corpus. First, a threshold is required to identify whether two keyframes are near-duplicate. From all the keyframes, a subset of 50 NDK pairs were labeled for estimating this threshold. By computing the similarities of the 100 keyframes based on Eq.(5.2), a 100×99 matrix is generated. The near-duplicate pairs are identified as positive while other pairs are taken as negative. The performance is evaluated with

Figure 5.7: PR-curve of the NDK detections on the labeled Trecvid dataset.

the Precision-Recall (PR) curve. Let D and T be the set of detected NDK pairs and groundtruth NDK pairs respectively. To be exact, the precision p measures the proportion of detected NDKs out of the groundtruth NDK pairs.

$$p = \frac{|D \cap T|}{|D|} \quad (5.12)$$

The recall r shows the proportion of correct NDK pairs detected in the dataset.

$$r = \frac{|D \cap T|}{|T|} \quad (5.13)$$

Fig. 5.7 shows the PR-curve of the NDK detections on the labeled dataset. The knee-point can be read from the curve, where $p = r = 0.92$. The similarity S in Eq.(5.2) corresponding to the knee-point is set as the keyframe detection threshold S_T . With S_T , we do the keyframe clique summarization for each video clip.

We adopt the compression ratio to evaluate the summarization result. Suppose M SKCs are formed out of a total number of N keyframes in one video clip, the compression ratio is defined as

$$CR = N/M. \quad (5.14)$$

Table 5.3: Keyframe Summarization of TRECVID 2005 and TRECVID 2006.

ID	TRECVID 2005				TRECVID 2006			
	Frame #	KF #	Clique #	CR	Frame #	KF #	Clique #	CR
1	50,930	557	129	4.32	111,469	248	60	4.13
2	50,935	480	163	2.94	56,771	74	16	4.63
3	102,701	615	224	2.75	111,715	225	23	9.78
4	49,116	485	205	2.37	111,981	336	51	6.59
5	104,852	992	258	3.84	111,872	267	26	10.27
6	50,879	417	146	2.85	111,776	176	22	8.00
7	50,921	637	128	4.98	30,574	256	104	2.46
8	50,827	377	133	2.83	57,963	460	142	3.24
9	50,905	336	141	2.38	57,917	633	180	3.51
10	44,873	449	145	3.10	112,055	535	90	5.94
11	50,931	504	164	3.07	112,084	1241	276	4.50
12	102,724	472	125	3.78	54,665	512	162	3.16
13	104,841	574	196	2.93	57,925	438	129	3.40
14	50,785	509	159	3.20	112,084	299	84	3.56
15	42,971	526	204	2.58	57,152	461	152	3.03
16	50,945	434	164	2.65	57,871	377	120	3.14
17	104,748	735	234	3.14	111,352	225	37	6.08
18	104,887	954	224	4.26	40,807	336	123	2.73
19	50,915	557	197	2.83	31,116	275	108	2.55
20	50,935	562	137	4.10	111,964	1,232	231	5.33
Total	1,321,621	11,172	3,476	3.21	1,621,113	8,606	2,136	4.03

Tab. 5.3 shows the compression ratios of the first 20 video clips from TRECVID 2005 and TRECVID 2006. The average compression ratios are 3.21 and 4.03 respectively. Due to the editing, motion and so on, there exist quite a number of Near-duplicate keyframes, some are due to the boundary detection method, others are due to similar scenes in different parts of the video clip. From the table, we see a significant amount of near-duplicates in these videos and our proposed method can effectively reduce the computation time and detect such near-duplicates. We also list the number of frames for each video (Tab. 5.3). From frames to SKCs, the average compression ratios are 380 and 759 for TRECVID 2005 and TRECVID 2006 respectively, which means SKCs

are very efficient for video representation.

Figure 5.8: Average near-duplicate retrieval performance comparison for different approaches over all queries. In this figure, “Temp_summ” means the method of segmentation-based temporal summarization. “Transitive_Closure” is the method based on transitive closure. “SKC_S” stands for our proposed Near-duplicate keyframe cliques based on keyframes from shot detection, while “SKC_T” stands for our proposed Near-duplicate keyframe cliques based on keyframes from dense sampling.

5.5.3 Evaluation of Keyframe Clique based Video Retrieval

In the second experiment, we apply SKCs for near-duplicate detection and removal in the web video dataset. All the keyframes from one video clip are partitioned into SKCs. The quasi-clique mining algorithm ensures each video clip, either query or candidate, to be represented with a set of weighted SKCs. We investigate our proposed video distance to compute the similarity between the query and the candidate videos. In order to show that our proposed SKCs do work for video summarization and retrieval, we compare it with different approaches, as well as different experiment settings.

Dense Sampling VS Shot Boundary Detection

As an effective form, a video clip is usually represented with a sequence of keyframes extracted by shot boundary detection or dense sampling. For shot boundary detection,

a series of interrelated consecutive frames are represented as a keyframe. However, most methods [31, 32] based on keyframes do not exploit the duration of the consecutive frames so that the importance of keyframes is not varied and all the keyframes are taken equally. Meanwhile, keyframes by dense sampling are selected with a certain steps. Thus a continuous action/event lasting longer will be represented with more keyframes. We evaluate our method under the two strategies above. Also we compare it with the baseline segmentation-based temporal summarization method, which merges every three consecutive keyframes into one group. Fig. 5.8 shows the average performance of the three approaches. We see that our proposed method outperforms the baseline method greatly for all the queries. Meanwhile, it shows the SKCs based on dense sampling stately are better than those based on shot detection, which supports our analysis above. The weights of SKCs based on dense sampling is more accurate than those on shot boundary detection.

Results comparison

We compare our method with other state-of-the-art methods: (i) SIG_CH [31], which measures video similarity based on the global features (color histogram)(ii) HIRACH [31], which is a two-level hierarchical method. To be exact, global features are first used to find the candidate videos and then local point matching is adopted to detect near-duplicate keyframes from different videos. As much as we know, the algorithm has achieved the highest result reported up to now. (iii) ALIGN [32], which is indeed an enhanced version of HIRACH. It accelerates the matching speed of two videos by considering temporal coherence. Tab. 5.4 shows the performance (average precision) of near-duplicate video detection under our proposed method and others. Our method achieves better results than HIRACH [31] using the same features (color histogram + PCA-SIFT). Note that both HIRACH and ALIGN have to do points

matching based on local features, which is very time consuming. We also compare our proposed SKC_T with the state-of-the-art method HIRACH in detail, see Fig. 5.9. From the figure, it shows that our proposed SKC_T outperforms HIRACH for most of queries. When the recall is high, the precision of SKCs is still promising. We also compare our SKCs with (iv) the transitive closure based method, which groups all the reachable keyframes into one clique. We compute the weights of each transitive closure and use the same video distance (Eq. 5.11) to compute the similarities between videos. The results are shown in Fig. 5.8 and Tab. 5.4, which prove that our SKCs are more advanced than the transitive closure based method.

Table 5.4: Performance Comparison for Near-Duplicate Video Detection between our proposed method and state-of-arts methods. The average precision is of all all queries over all recall levels.

Approaches & details	AP (%)
SKC_T NDK Clique, dense sampling, our video distance	97.2
SKC_S NDK Clique, shot boundary detection, our video distance	95.4
Transitive_Closure Transitive Closure among keyframes, our video distance	92.0
Temp_Summ Each group containing three continuous keyframes	85.9
SIG_CH Keyframe matching, Color histogram	89.2
HIRACH Two level keyframe matching	95.2
ALIGN Keyframe matching with alignment	95.1

Figure 5.9: PR-curves of 24 queries for HIRACH and our proposed SKC_T method.

Sampling of SKCs for video retrieval

One of the important merits of our proposed SKCs is that it is scalable for video retrieval. The SKCs are ranked and give a clue to select the most important ones for faster video retrieval. The weight represents the duration and frequency of the SKCs occurring in one video. The larger the weight is, the more important it plays in the whole video. To speed up, one can use the top SKCs. In Fig. 5.10, we show the APs corresponding to different sampling rates (from 100% to 10%) of SKCs. As a comparison, we also show the results by random sampling method. We observe that the performance (95.82%) is very high even we use the top 10% of NDK cliques, while the performance is poor for random selection. Comparing to random selection strategy, in which the average performance decreases sharply when the sample rate becomes smaller, the performance of the top SKCs keeps stable all the time. It indicates that the top SKCs hold the most important information of the video at any sampling rate, which is very helpful for fast video management and search, especially when the data size is tremendous.

Figure 5.10: Average near-duplicate retrieval performance for different sampling rates of SKCs. The red curve is the result using SKCs with different sampling rates by their weights ranking, while the blue one is from random selection.

SKCs for large scale video retrieval

Compared with the state-of-the-art methods, our method has shown excellent performances. In this section, we further evaluate it in a larger-scale video retrieval task. As there is no large-scale testbed published up to now, we still use the above near-duplicate web video dataset. The difference is that for each query, we search its near-duplicates from all the 12,789 reference videos. To show the effectiveness and scalability of our approach, we use two sampling rates (100% and 10%). The PR-curves are computed and shown in Fig. 5.11. Though with a large-scale reference set, our approach still achieves a promising result. Exactly, the average precision is up to 94.66% when all SKCs are adopted for retrieval. And the performance is still encouraging (92.71%) when the sampling rate is only 10%. Comparing to the Fig. 5.10, the AP drops is mainly caused by the dropping of precisions corresponding to large recall values ($> 80\%$), which are vulnerable to be effected by some challenging near-duplicate videos. In a larger-scale dataset, the near-duplicate videos with a large variation from the query video are sorted behind more noises videos thus resulting in the slightly dropping of the perfor-

mance. However, these challenging near-duplicate videos are of a small number and the performance remains high and promising in a larger reference set.

Figure 5.11: The PR-curves for two sampling rates of SKCs with a large-scale reference set. The average precisions for sampling rate 100% and 10% are 94.66% and 92.71% respectively.

5.6 Conclusions

In this chapter, we introduce a stratification-based approach to describe videos. By our proposed quasi-clique mining algorithm, a series of Stratification Keyframe Cliques (SKCs) with different weights are generated to represent each video. The representations are compact and informative, which not only speed up the retrieval process, but also maintain a high accuracy. What’s more, the speed will be further accelerated when only the top SKCs are used. Due to its efficiency and effectiveness, our proposed algorithm is promising for web video representation and retrieval, especially for large scale datasets.

Chapter 6

Biologically-Inspired Spatially-Coherent Pyramid Matching

6.1 Introduction

High level image representation is crucial for image retrieval and classification. Among different methods, “Bag-of-Words” model [19] is a widely-used approach for generic image categorization and retrieval, which treats the whole image as a document and represents it as a histogram by assigning all the descriptors of local covariant regions to “visual word” frequency. The weakness of this model is that it takes all the features into account, including the “noises”. In addition, it also brings in variations when the proportion of object size or the number of objects are different between images. For example, the first image pair in Fig. 6.1(a), we identify that the image belongs to the daffodil class if we can detect there exist daffodil flowers, no matter how many of them exist in the image. Different from BOW models, parts-based models [33, 14, 41] focus

Figure 6.1: Examples of image pairs. Due to the scale, rotation and position variations, the image in the same class is not region-by-region correspondingly.

on the discriminative parts within an image. If some of the object parts can be detected in the image, one can conclude that such an object exists in the image. Though existing part-based models have their limitations, such as relying on the representative power of each part, they give us an important indication that the image can be efficiently represented with partial content instead of the global summations of all parts.

Both objects and scenes have their own structures, which is why human brain can easily identify objects that have been seen before. Many researchers [26, 21] have proved that spatial information plays an important role in image representation. Lazebnik et al. [24] extended the Pyramid Matching Kernel (PMK) [25] by incorporating the spatial information of local regions. One image is partitioned into increasingly fine sub-areas and form a hierarchical tree representation. This Spatial Pyramid Matching (SPM) method implicitly assumes a fixed tree structure. That is, the same sub-regions of images from the same class share the same visual properties, which is the reason why SPM works exceptionally well for scene classification. However, for object classification the assumption may not hold as the object may have different scales and locations. See Fig. 6.1 for example. Due to the scale, rotation and position variations, the images

in the same class (Fig. 6.1 (a), (b), (c) respectively) do not show region-by-region correspondences. It is therefore not appropriate to compare the image by reshaping the hierarchical tree in a plain vector.

In fact, what we observe is that the pyramid representation shows not only the rigid hierarchical structure of the image, but also the correspondences between subregions within a layer and even crossed layers. Their dependency information in each layer or even crossed layer is relaxed but robust for most images. In this chapter, our objective is to formulate the image similarity as a hierarchical tree similarity problem. The motivation behind the work reported in this chapter is to build the hierarchical image representation by a bottom-up maximum pooling strategy, which uses the saliency characteristics in low-layer regions to describe high layer regions. The main advantage is that it is robust to clutter and the image is not dominated by large areas of similar pattern. What's more, we incorporate the invariant properties among the hierarchical tree structure. That is, spatial(within-layer) and crossed-layers dependencies between close regions are exploited.

The rest of the chapter is organized as follows. We review the related work in Section 6.2. Section 6.3 gives the details of building an image representation by a biologically-inspired bottom-up strategy. In Section 6.4, we extend the pyramid matching of images by incorporating correspondences between regions (thereby achieving spatially-coherent). The experiment results are reported and compared with other methods in Section 6.5. Finally, we conclude the contributions of this chapter in Section 6.6.

6.2 Related Work

“Bag-of-Words” model [19] is an effective approach for generic image categorization and retrieval. Under this model, images are treated as documents by assigning descriptors of local covariant regions to “visual words”. Each image is then represented by a histogram of word frequency. However, this model has two major weaknesses. One is that it is not clutter-tolerant due to all features being taken into consideration. The other is that the BoW model loses all the spatial information because of the assumption of independent features.

Researchers [101, 102] working with biological techniques adopted the max pooling procedure to minimize the clutter problem. It can be formalized mathematically as an operation that returns the largest of its inputs. Thus a neuron that performs a maximum operation on its pooled inputs will respond to the strongest of its inputs only and will not be affected by other weaker inputs. The max pooling procedure is well established by biophysical evidence in visual cortex (V1) [102] and is empirically justified by many algorithms applied to image categorization. Yang et al. [103] adopt the max pooling statistics for descriptor-based sparse codes. That is, in a subregion with various features. Using the maximum stimuli after sparse coding of each feature to represent the region. However, such representations rely heavily on the discriminative nature of features.

On the other hand, several methods have been proposed to improve the BoW model by capturing the spatial arrangement of visual words. Lazebnik et al. [24] extended the Pyramid Matching Kernel (PMK) [25, 56] by incorporating the spatial information of local regions. Two images are partitioned into increasingly fine sub-areas and PMK is used to compare corresponding sub-regions. This Spatial Pyramid Matching (SPM) method implicitly assumes matching occurs between corresponding sub-regions in the

same layer and location. It is therefore not scale or rotation invariant, as well as sensitive to the change in position. Also by utilizing spatial information of local regions, Savarese et al. [26] used *correlagram* to measure the distribution of the proximities between all pairs of visual words and used for category classification, but the method is not scale invariant. Proximity Distribution Kernels (PDK) was proposed by Ling et al.[21], which captures the rank information about the proximities between visual words corresponding to two interest points. However, Ling’s method relies heavily on the robustness of detectors and missing some interesting points will change the rank information. Based on SPM, Xu et al [89] proposed a framework called Spatially Aligned Pyramid Matching (SAPM) to resolve the problem of position and scale changes. By using Earth Mover’s Distance (EMD), pairwise similarity between blocks are examined and then multiple alignment hypotheses are resolved using EMD in another layer. Due to the huge computation cost of multi-EMD, this algorithm is inefficient and cannot be applicable for real-time online process.

6.3 Biologically-Inspired Pyramid Matching

Traditional bag-of-words model use the mean of the visual words in the whole region. Such methods take all features including “noise” points into consideration. As the spatial pyramid matching is based on the BoW model, it is also not clutter-tolerant. In this section, we propose our method to build a hierarchical tree using a biological-inspired max pooling strategy. Usually, an object is formed of various parts. However, to differentiate one object from another, only the distinctive parts are required to form the object model, while the other parts of image can be neglected. For example, we can judge whether an image is in the face category when we successfully detect the patches of eyes, nose and mouth. Without prior knowledge, it is better to build the

structure by bottom-up procedure in the cognitive process of understanding the image.

Figure 6.2: Comparison of biological-inspired spatial pyramid and the general spatial pyramid. The blue circle and red square stand for two different patterns in the image. The red numbers in BiScPM indicate the maximum-pool value for the region in the respective layer and location.

6.3.1 Subregions to describe the image structure

Currently, most of the methods take the single feature as the basic element. However, single feature is limited due to its descriptive power. It is sensitive to noises and object deformations. Instead of feature points, we use subregions with a number of features as the element of an image, which contain sufficient statistics of the local region. To be exact, the image is divided into small subregions. Each subregion contains one or limit parts of the object/scene. We mine the image information based on these subregions. In order to get sufficient statistics, the dense sampling patches are adopted. Each patch is then describe by the SIFT feature [5]. The SIFT features are quantized to a fixed-sized visual words and the subregion is represented by the bag-of-words histogram. To be exact, we represent it as

$$H = (h_1, h_2, \dots, h_K), \tag{6.1}$$

where K is the number of visual words generated by the clustering algorithm.

6.3.2 Maximum spatial pooling

For each upper layer region, it contains several lower subregions. In SPM model, it is the mean of the lower regions. However, as we analyzed above, the mean operation is not robust to clutter. All the noise features will be taken into consideration. Meanwhile, maximum operation has been proven to be clutter-tolerant. This max pooling procedure is well established by biophysical evidence in visual cortex. Given the subregion representation $H_{i1}, H_{i2}, \dots, H_{iM}$, The upper layer representation will be

$$H_i = \max\{H_{i1}, H_{i2}, \dots, H_{iM}\} \quad (6.2)$$

That is, for each bin in the histograms of lower layer regions, the maximum value is adopted as the bin value in the higher layer. The maximum operation is in fact using the saliency characteristics of object/scene to represent the whole image. See Fig. 6.2 for illustration.

In order to handle the problem that one structure might be partitioned into different subregions, we adopt the overlapping strategy. See Fig. 6.3 for illustration. That is, the subregions are specified with a fixed smaller step comparing to their size. For each visual word, the value of the higher layer is equal to the max of the values in the lower subregions. The overlapping strategy ensures most of the important patterns are not segmented in different regions. What's more, the features will not be counted repeatedly because of the maximum spatial pooling procedure. For SPM, it cannot well-handle object that cut-across the fixed subregions in the lower layer.

6.4 Spatially-coherent Pyramid Matching

The pyramid representation shows the hierarchical properties of the image. In this section, our objective is to formulate the image similarity as a modified hierarchical

Figure 6.3: An illustration of how object pattern that is not aligned to subregions is handled by SPM & our BiScPM. The red box is the selected subregion. For BiScPM, maximum spatial pooling with overlapping regions will generate a better matched subregion.

tree similarity. Due to the transition, rotation and scale variances, images do not follow a rigid hierarchical tree structure. Therefore, we do not treat image matching as the traditional hierarchical tree matching. Instead, we incorporate the invariant properties within the structure of hierarchical tree.

6.4.1 One-to-one subregion matching

In order to better explain the problem, we use $G = (V, E)$ to represent the hierarchical tree, which in fact can be seen as a directed graph. V is the vertex corresponding the regions and E is the edge which indicate if two regions are co-occurrent. In details, we link not only the nodes between children and parent, but also the sibling nodes which are spatial neighbors.

Without loss of generality, we represent an image with the hierarchical regions, noted as $V = \{v_1, v_2, \dots, v_N\}$, in which N is the total number of subregions from all the layer L and $N = L(L - 1)/2$. Each v_n is a histogram representation by maximum pooling. Given two images V_p and V_q , we measure the feature similarities of all the

subregions. That is,

$$s_{ij} = f(v_{pi}, v_{qj}) \quad (6.3)$$

The equation shows the visual similarity of different regions from different layers and spatial locations.

Any two images have the same cardinality, that is they have the same number of subregions. On the other hand, the correspondence can be shown to exist between any two sets that have the same cardinality. Our objective is to maximize the following equation,

$$\begin{aligned} & \text{maximize} \quad \sum_{i,i'} w_{ii'} s_{ii'} \\ & \text{subject to} \quad w_{ii'} \in \{0, 1\}, \quad 1 \leq i \leq N, \quad 1 \leq i' \leq N \\ & \quad \sum_{i'=1}^N w_{ii'} \leq 1, \quad 1 \leq i \leq N \\ & \quad \sum_{i=1}^N w_{ii'} \leq 1, \quad 1 \leq i' \leq N \end{aligned} \quad (6.4)$$

From Eq. 6.4, we see that there are one-to-one correspondences between the two images. That is, each subregion of the first image is paired with one and only one subregion of the second image, while each subregion of second image is paired with one and only one subregion of the first image, and none of the subregions from either set are left unpaired.

Table 6.1: Notation used throughout this algorithm of Hungarian Algorithm.

symbol	description
G_b	bipartite graph
l	the vertex labeling function
G_l	equality graph w.r.t l
E_l	edge in equality graph.
	$E_l = \{(x, y) : l(x) + l(y) = w(x, y)\}$
$N_l(u)$	neighbor of $u \in V_b$

To solve the one-to-one matching problem, we follow the Hungarian Algorithm [104]. That is, given a weighted bipartite graph $G_b = (V_b, E_b)$, the max-weight matching M is to be found, in which every vertex is adjacent to some edge in M . The procedure of the algorithm is shown in Algorithm 2. The notation used throughout this algorithm is summarized in Tab. 6.1.

Algorithm 2 Hungarian Algorithm

- 1: $G_b = (V_b, E_b)$
 $//V_b = X \cup Y$ and $X \cup Y = \phi$
 $//E_b \subseteq X \times Y$
 - 2: Generate initial labeling l and matching M in E_l
 - 3: If M perfect, stop
 Otherwise, pick free vertex $u \in X$.
 Set $S = u$, $T = \phi$
 - 4: If $N_l(S) = T$, update labels (forcing $N_l(S) \neq T$)
 $\alpha_l = \min_{s \in S, y \notin T} \{l(x) + l(y) - w(x, y)\}$
 $l'(v) = \begin{cases} l(v) - \alpha_l & \text{if } v \in S \\ l(v) + \alpha_l & \text{if } v \in T \\ l(v) & \text{otherwise} \end{cases}$
 - 5: If $N_l(S) \neq T$, pick $y \in N_l(S) - T$
 If y free, $u - y$ is augmenting path.
 Augment M and go to 3.
 If y matched, say to Z , extend alternating tree:
 $S = S \cup \{z\}$, $T = T \cup \{y\}$. Go to 4.
-

After matching the subregions of different scales and location from two images, we measure the similarity of the two images as follows:

$$S = \sum_{i, i'} w_{ii'} s_{ii'} \quad (6.5)$$

6.4.2 Spatially-coherent region correspondence

The correspondence between patches is an important factor in the hierarchical tree. Compared to the “global” tree structure, the correspondence between regions can be seen as “local”. The nodes are not isolated from each other. For two pairs $M_i = (i, i')$

Figure 6.4: Spatially-coherent Pyramid Matching. The subregions with colored outline are matched pairs. We see that if the matches are correct, the link between them are corresponding in the two image. Illustration is best seen in the color pdf version.

and $M_j = (j, j')$, if i and j have a link and i' and j' also have a link, they give extra support for matching the tree. To well explore the dependence, we define two types of correspondences:

1. Correspondence between parent and children. As the child node is subregions of the parent, it shows a dependence in the scale space (crossed layers). See the red lines in Fig. 6.4.
2. Correspondence between siblings. The sibling correspondences show the spatial dependence within a layer. See the blue lines in Fig. 6.4.

To indicate the correspondence, we have,

$$\delta_{ij} = \begin{cases} 1 & \text{if there is a link between } i \text{ and } j \\ 0 & \text{otherwise} \end{cases} \quad (6.6)$$

To show the coherence of two pairs, we use the following equation:

$$c_{ii'jj'} = \delta_{ij}\delta_{i'j'}d_{ii'}d_{jj'} \quad (6.7)$$

That means in the matched pairs (i, i') and (j, j') , $c_{i,j}$ is non-zero if and only if there are correspondence links between i and i' as well as j and j' .

Finally, the similarity of two images can be measured as follows:

$$\hat{S} = \sum_{i,i'} w_{ii'} d_{ii'} + \sigma \sum_{i,i',j,j'} c_{ii'jj'} w_{ii'} w_{jj'} \quad (6.8)$$

In the above objective function, the first part measures the similarities of the subregion pairs. And the second part is to evaluate the correspondence of different matched pairs, either spatial or scale dependencies. The parameter σ is the trade-off between the two parts.

6.5 Experiments

Figure 6.5: Top- k IND retrieval performance of our proposed method and relevant methods on the Columbia dataset and CityU dataset. Note: the accuracy in (b) sometimes decreases with the rank due to more than one NDKs for each keyframe.

In our experiment, we adopt the dense sampling strategy with 16×16 patch size. For each patch, we use the SIFT [5] representation. The general k-means clustering method is used and a vocabulary of 500 visual words is generated. To measure the similarity between subregions, we take the RBF kernel as the function f in Eq. 6.3. That is,

$$f(H_1, H_2) = \exp\left(-\frac{1}{\alpha} D(H_1, H_2)\right) \quad (6.9)$$

Here we adopt the fixed $\alpha = 2$. And D is the χ^2 -distance, which has been proven to be a very effective similarity measurement of histogram-based representation[88]. Therefore, given two histograms $H_1 = (x_1, \dots, x_n)$ and $H_2 = (y_1, \dots, y_n)$, the χ^2 -distance is defined as

$$D(H_1, H_2) = \frac{1}{2} \sum_{i=1}^n \frac{(x_i - y_i)^2}{(x_i + y_i)} \quad (6.10)$$

For the retrieval problem, we use Eq. 6.8 to measure similarity of query images and the candidates. For classification problem, we use it as the kernel and the LIBSVM [105] is used to train the model. Parameters are learned by n -fold ($n = 5$) cross validation.

6.5.1 Image Near-Duplicate Retrieval

In this section, we evaluate our proposed method on two Image Near-Duplicate(IND) datasets which were extracted from the keyframes of TRECVID corpus [78]. The Columbia dataset¹ is provided by [1], which are the keyframes extracted from TRECVID 2003 corpus. Each dataset includes 150 IND pairs (300 images) and 300 non-duplicate images. The size of all the images are 352×264 . The second dataset CityU dataset² is a much larger dataset from TRECVID 2004 corpus, with 7006 keyframes including 3388 near-duplicate image pairs. In this dataset, there exist more than one near-duplicate images for some keyframes. To compare with other state-of-art method, we adopt the same experiment setup for IND retrieval. All IND pairs are used as queries to evaluate performance. For each query, we calculate the similarity between the query image and all the other images. A ranked list is then produced according to their similarity to a query. To evaluate the retrieval performance of our proposed BiScPM and comparison with other state-of-art methods, we estimate the probability

¹<http://www.ee.columbia.edu/ln/dvmm/researchProjects/FeatureExtraction/NearDuplicateByParts/INDDetection.html>

²<http://vireo.cs.cityu.edu.hk/webvideo/>

Table 6.2: Top-1 retrieval performance comparison of our proposed method, SAPM, SPM and TPM on both Columbia Dataset. For SPM, TPM and SAPM, the first number is the performance using only one layer, and the second is the highest results using multilayer with different fusion methods.

Methods	Columbia Dataset(%)
SPM	76.3/78.0
TPM	74.7/75.7
SAPM	79.0/80.7
Our BiScPM	88.7

of the successful top- k retrieval $P(k)$ as follows

$$P(k) = \frac{Q_c}{Q} \tag{6.11}$$

where Q_c is the number of queries that rank their INDs within the top- k position, and Q is the total number of queries.

We compare our proposed BiScPM with other state-of-the-art methods: (i) the NIM [85] (ii) the S^3VM [85], (iii) the LIP-IS+OOS [2], (iv) the OOS-SIFT approach [90] (v) the Visual Keywords(VK) methods [90] and (vi) color moment [91] or color histogram [1] based retrieval. Figure 6.5 shows the results of these methods for top- k IND retrieval on the Columbia dataset and cityU dataset where k changes from 1 to 30. From the figure, we can see that our BiScPM significantly outperforms the other methods in terms of accuracy.

We also compare our proposed approach with Spatial Pyramid Matching (SPM)[24], Temporal Pyramid Matching (TPM) and Spatially Aligned Pyramid Matching (SAPM)[89], which are related methods to ours as they also incorporate some kind of spatial information. Table 6.2 shows the top-1 retrieval performance on the Columbia Dataset. Our results significantly outperform the other three approaches, which shows that our strategy of using bottom-up spatial maximum pooling and spatially-coherent

correspondence exploring in the hierarchical tree did provide more discriminant information for determining IND.

6.5.2 Event Classification

Our algorithm is tested on the sports event dataset³ [106]. This dataset was collected from the Internet, and was later annotated thoroughly by Lotus Hill Research Institute. There are 8 sports categories in the dataset: bocce, croquet, polo, rowing, snowboarding, badminton, sailing, and rock climbing. The number of images in each category varies from 137 to 250. In each image, there are always one or more athletes, and the poses and sizes of these athletes differ significantly.

In this work, we take the same classification protocol setup as in [106]: For each event class, 70 randomly selected images are used for training and 60 of the remaining images are used for testing. The training data and testing data are split randomly for 5 times and average accuracy is reported. We compare our method with other corresponding methods.

The experiment results are reported in Table 6.3. A single scale for dense sampling points extraction is adopted. In the sports event dataset, our proposed method achieves an accuracy of 84.79 ± 0.53 . The pyramid matching kernel (PMK) [25] has an accuracy of 78.84 ± 0.55 . Comparing with other state-of-arts methods reported on this dataset, our proposed method also show better performance. The result in [106] is 73.4, which is based on scene environment classification and object categorization. The spatial PACT, which adopt a spatial pyramid of PACT feature, achieved 78.50 accuracy. We also compared with Histogram Intersection Kernel (HIK) [107]. If both methods use only one scale of dense sampling points, our proposed method greatly outperforms HIK

³http://vision.stanford.edu/lijiali/event_dataset/

Table 6.3: Comparison of classification results on Princeton 8-sports event dataset. Note HIK adopted one scale of dense sampling points, while HIK2 adopted multi-scales of dense sampling points as well as Sobel images.

Methods	Accuracy(%)
PMK[25]	78.84±0.55
scene/object[106]	73.4
Spatial PACT[108]	78.50
HIK[107]	77.54
HIK2[107]	84.21
Our BiScPM	84.79±0.53

(77.54). And our result is even better to HIK2 (84.21), when it adopted 5 scales of dense sampling points, as well as the Sobel images.

Fig. 6.6 shows the confusion matrix for our proposed method. In the confusion table, the rows represent the ground truth categories of events and the columns represent the predict labels. Compared with the confusion matrix given in [106, 108], our BiScPM outperforms theirs for most categories. Although it is still difficult to separate the bocce and croquet sport events due to very similar background, our experiment results greatly reduce the error rate, which shows the representative and discriminative power of our proposed method.

6.5.3 Webvideo Retrieval

The third experiment is to gauge the performance of our algorithm for detection/removal of near-duplicate web-video clips. The dataset⁴ is provided by Wu [31] and it was crawled from YouTube, Google Video, Yahoo Video. In all, there are a total of 12790 web-video clips. The benchmark dataset is organized into 24 search queries, where each search query has a ground truth of the total number of web-video

⁴<http://vireo.cs.cityu.edu.hk/webvideo/>

Figure 6.6: Confusion Matrix of Princeton sports event dataset. The average accuracy is 84.79% for all the 8 categories.

clips and its subset of near-duplicate video clips. There are two steps for webvideo retrieval. First, the near-duplicate frames are detected in each video. Then each group of near-duplicate frames generates one biological-inspired spatially-coherent pyramid by considering the same subregions from all the frames. The similarity of the video is the summation of all the groups of near-duplicate frames in the video.

We compare our proposed BiScPM with other state-of-the-art methods: (i)the SIG_CH [31], which measured video similarity based on the global features(color histogram)(ii) the HIRACH [31], which is a two-layer hierarchical method. To be exact, the global features are first used to find the candidate near-duplicate videos and then local point matching is adopted to detect near-duplicate keyframes. As far as we know, the algorithm achieved the highest result reported today. (iii) ALIGN [32], which is an enhanced version of HIRACH. It accelerates the matching speed of two videos by

considering temporal coherence. Tab. 6.4 show the performance (average precision) of near-duplicate detection using our BiScPM as well as other different methods. Our method achieves the best performance among all the results reported today. We also compare the precision of our BiScPM with the state-of-arts methods with different recalls, see Fig. 6.7. From the figure, it shows that our BiScPM outperforms HIRACH at all recall levels. What’s more, we see that the precision is still promising when the recall is high.

Table 6.4: Performance Comparison for Near-Duplicate Video Detection between our proposed BiScPM and state-of-arts methods. The average precision is of all all queries over all recall levels.

Approaches & details	AP (%)
SIG_CH [31] Keyframe matching, Color histogram	89.2
HIRACH [31] Two level keyframe matching	95.2
ALIGN [32] Keyframe matching with alignment	95.1
Our BiScPM biologically-inspired spatially-coherent pyra- mid matching	97.9

6.6 Conclusion

This chapter presents a biologically-inspired pyramid strategy for image representation, as well as the spatially-coherent pyramid matching for image similarity measuring. By applying max pooling of salient characteristics from the lower-layer subregions, the

Figure 6.7: Comparison of average Precision-Recall curves of 24 webvideo queries with different methods.

representation of higher-layer region has been shown to be more clutter-tolerant. By matching the regions at different layers and incorporating geometry correspondences between matched regions, the results enforced our claims of scale and position invariant. Our experiments has shown that the proposed BiScPM can be applied in the area of image near-duplicate detection, event classification and webvideo retrieval. In general, BiScPM can be used for other retrieval tasks and classification problems. Results on the various datasets show that our BiScPM method achieved significant improvement in every experiment.

Chapter 7

Conclusion

7.1 Research Summary

The main focus of this thesis is on algorithms of local dependencies for image/video representation. In this context, the following contributions have been made:

1. A coherent phrase model is proposed for image near-duplicate retrieval. Different from the standard BoW representation, this model represents every local region using multiple descriptors and enforces the coherency across multiple descriptors for every local region. Two types of visual phrase (feature coherent phrase and spatial coherent phrase) are designed to represent feature and spatial coherency. Both coherency are utilized without increasing the computational complexity. This model, although simple, improves the matching accuracy by reducing the number of false matches and preserves the matching efficiency because of the sparsity of the representation. Instead of mining the association of multiple regions which increases the complexity, the quantization error of BoW is reduced by the coherency in the visual phrase of every single region in our method. Both

effectiveness and efficiency of the proposed method have been proved on multiple benchmarks built from TRECVID corpus for IND retrieval as well as a benchmark for object retrieval. The proposed technique can be generally applied on different variants of the BoW model.

2. Local dependencies in spatial-scale space (S-Cube representation) are analyzed to form a new representation, which incorporates not only the appearance feature information, but also the spatial and scale co-occurrence information. Through the integration of local dependencies over neighboring spatial locations and multiple scales, the proposed S-Cube method is invariant to spatial transformation and a wider range of scale change. Experiments on three IND datasets have shown that S-Cube is a promising approach for IND retrieval. In addition to improving retrieval accuracy, a faster retrieval speed has shown that our proposed S-Cube has great potential in Near-Duplicate image retrieval and IND assisted video retrieval.
3. New video representations are introduced with a stratification-based approach. By our proposed quasi-clique mining algorithm, a series of Stratification Keyframe Cliques (SKCs) with different weights are generated to represent each video. The representations are compact and informative, which not only speed up the retrieval process, but also maintain a high accuracy. What's more, the speed will be further accelerated when only the top SKCs are used. Due to its efficiency and effectiveness, our proposed algorithm is promising for web video representation and retrieval, especially for large scale datasets.
4. We present a biologically-inspired pyramid strategy for image representation, as well as the spatially-coherent pyramid matching for image similarity measuring. By applying max pooling of salient characteristics from the lower-layer

subregions, the representation of higher-layer region has been shown to be more clutter-tolerant. By matching the regions at different layers and incorporating geometry correspondences between matched regions, the results enforced our claims of scale and position invariant. Our experiments has shown that the proposed BiScPM can be applied in the area of image near-duplicate detection, event classification and webvideo retrieval. In general, BiScPM can be used for other retrieval tasks and classification problems. Results on the various datasets show that our BiScPM method achieved significant improvement in every experiment.

In summary, we proposed four approaches in this dissertation, that is, Coherent Phrase Model (FCP/SCP), S-Cube, SKC and BiScPM. They all incorporate the local dependencies in order to enrich the representative power. However, local dependencies are exploited in different ways. For coherent phrase model, different attributes are considered for a single point, while both the neighborhood and scale space are studied in S-Cube. For SKC, the local dependencies are mainly focus on temporal domain. And both the local dependencies and global consistencies are incorporated in BiScPM.

7.2 Future Work

Till now we have investigated and developed a number of local dependencies methods for recognition and retrieval. Although encouraging results have been achieved in terms of both detection accuracy and efficiency, there are still some limitations in our proposed models. Exactly speaking, all the coherent/co-occurrence pairs are taken equally in either coherent phrase model or S-Cube, which are used as general attributes for all the classes of objects. That is, these models failed to tackle the unique object-specific characteristics. In the SKC model, the threshold t for quasi-clique is empirically chosen. In the BiScPM model, the subregions are obtained by grid partition of image,

which make it lack of semantic meanings. Our proposed algorithms can be further extended into more general cases, and better performance is reachable. In the future, we plan to extend the current work in the following directions.

1. Multi-feature Dependencies Exploiting: An object consists of different characteristics. It is insufficient to describe it using only one features. To make an accurate description, more features and dependencies are required. For example, human lips are notable red coloring, below the nose, and with a philtrum on the upper lip. However, exploring the dependencies among multi-features is challenging. How to find the stable dependency features is an important issue to be further studied.
2. Attribute Mining: Instead of mapping low-level image features directly to object category labels, models using visual attributes are proposed recently. The attributes are visible properties such as “with eyes” or “green colors”, which provide a semantic meaning in various issues. Meanwhile, relative information has been explored in vision in a variety of ways. WordNet-based information is exploited to specify a semantic-distance sensitive classifier. And explicit similarity-based supervision is adopted to share training instances for categories with limited or no training instances. Our proposed local-dependency-based representations go beyond low level features, which cover larger local regions and their inter-relations and provide a discriminative high-level descriptions. we believe that many of them share an explicit attribute/structure. By mining them, we may obtain the object-dependent, class-dependent and even cross-class patterns, which are definitely helpful in many different vision tasks.
3. Build Object Models with Features and Their Dependencies: The hierarchical structure is nature for objects. For example, human beings consist of different

organs. Organs consist of tissues. And tissues consist of cells. Among the hierarchical structures, the spatial layout also plays an important role. Without supervision, it is reasonable to build an object using a bottom-up strategy. Salient children parts with their invariant interactions are adopted to form higher parts. And the higher parts and their dependencies are used to build even higher level parts. It is an interesting and challenging topic to explore the invariant properties, either features or dependencies, in the hierarchical structures.

Publications List

Journal Papers

1. **Xiangang Cheng**, Liang-Tien Chia, “ Stratification-based Keyframe Cliques for Effective and Efficient Video” IEEE Transactions on Multimedia 13(6): 1333-1342 (2011)
2. **Xiangang Cheng**, Yiqun Hu, Liang-Tien Chia, “Exploiting Local Dependencies with Spatial-Scale Space (S-Cube) for Near-Duplicate Retrieval Recognition,” Journal of Computer Vision and Image Understanding (CVIU), 2011
3. Shenghua Gao, Liang-Tien Chia, **Xiangang Cheng**, “ Web image concept annotation with better understanding of tags and visual features”, J. Visual Communication and Image Representation 21(8): 806-814 (2010)
4. Yiqun Hu, **Xiangang Cheng**, Liang-Tien Chia, Xing Xie, Deepu Rajan and Ah-Hwee Tan, “Coherent Phrase Model for Efficient Image Near-Duplicate Retrieval,” IEEE Trans. on MultiMedia, 2009

Conference Papers

1. **Xiangang Cheng**, Liang-Tien Chia, “Spatially-coherent Pyramid Matching Based on Max-pooling,” ACM Multimedia 2011: 1445-1448.
2. **Xiangang Cheng**, Jingdong Wang, Liang-Tien Chia, Xian-Sheng Hua, “Learning to combine multi-resolution spatially-weighted co-occurrence matrices for image representation,” IEEE International Conference on Multimedia and Expo (ICME 2010). Singapore. July 19-13, 2010
3. **Xiangang Cheng**, Liang-Tien Chia, “Stratification-based Keyframe Cliques for Removal of Near-Duplicates in Video Search Results,” Proceedings of 11th ACM SIGMM International Conference on Multimedia Information Retrieval (MIR’2010), Philadelphia, Pennsylvania, March, 2010
4. Shenghua Gao, **Xiangang Cheng**, Liang-Tien Chia, “Discovering Class-Specific Informative Patches and Its Application in Landmark Characterization,” The 16th International Multimedia Modeling Conference (MMM2010), Chongqing, China. January 2010: 218-228

5. **Xiangang Cheng**, Yiqun Hu, Liang-Tien Chia, "Hierarchical Word Image Representation for Part-based Object Recognition," Proceedings of IEEE International Conference on Image Processing (ICIP) 2009
6. Shenghua Gao, **Xiangang Cheng**, Huan Wang, Liang-Tien Chia, "Concept Model-based Unsupervised Web Image Re-ranking," Proceedings of IEEE International Conference on Image Processing (ICIP) 2009
7. Shenghua Gao, Liang-Tien Chia, **Xiangang Cheng**, "Understanding tag-cloud and visual features for better annotation of concepts in NUS-WIDE dataset," WSMC '09: Proceedings of the 1st workshop on Web-scale multimedia corpus, October 2009
8. **Xiangang Cheng**, Yiqun Hu, Liang-Tien Chia, "Image Near-Duplicate Retrieval Using Local Dependencies in Spatial-Scale Space," Proceedings of the 16th annual ACM international conference on Multimedia(MM'2008), Vancouver, British Columbia, Canada. October 2008

Bibliography

- [1] Dong-Qing Zhang and Shi-Fu Chang. Detecting Image Near-Duplicate by Stochastic Attribute Relational Graph Matching with Learning. In *Proceedings of the 12th annual ACM international conference on Multimedia*, pages 877–884, New York, NY, USA, October 2004.
- [2] Wan-Lei Zhao, Chong-Wah Ngo, Hung-Khoon Tan, and Xiao Wu. Near-Duplicate Keyframe Identification with Interest Point Matching and Pattern Learning. *IEEE Transactions on Multimedia*, 9(5):1037–1048, August 2007.
- [3] Chong-Wah Ngo, Wan-Lei Zhao, and Yu-Gang Jiang. Fast Tracking of Near-Duplicate Keyframes in Broadcast Domain with Transitivity Propagation. In *Proceedings of the 14th annual ACM international conference on Multimedia*, pages 845–854, Santa Barbara, CA, USA, October 2006.
- [4] J.M.O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. *British Machine Vision Conference*, 72, 2002.
- [5] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.
- [6] Yan Ke and Rahul Sukthankar. PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 511–517, Washington, DC, June 2004.
- [7] Krystian Mikolajczyk and Cordelia Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, October 2005.
- [8] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded Up Robust Features. In *European Conference on Computer Vision*, Graz, Austria, May 2006.

-
- [9] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. A Sparse Texture Representation Using Local Affine Regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, August 2005.
 - [10] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. *Advances in Neural Information Processing Systems*, pages 831–837, 2000.
 - [11] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1:511–518, 2001.
 - [12] M. Marin-Jimenez and NP de la Blanca. Empirical Study of Multi-Scale Filter Banks for Object Categorization. *Proc. Int. Conf. on Pattern Recognition*, 2006.
 - [13] J.L. Shih and L.H. Chen. Color Image Retrieval Based on Primitives of Color Moments. *Proceedings of the 5th International Conference on Recent Advances in Visual Information Systems*, pages 88–94, 2002.
 - [14] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2, 2003.
 - [15] F.F. Li and P. Perona. A Bayesian Hierarchical Model for Learning Natural Scene Categories. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 524–531, 2005.
 - [16] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.
 - [17] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004.
 - [18] D. Gokalp and S. Aksoy. Scene classification using bag-of-regions representations. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Beyond Patches Workshop, Minneapolis, Minnesota, June, 23, 2007*.
 - [19] Josef Sivic and Andrew Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proceedings of IEEE International Conference on Computer Vision*, volume 2, pages 1470–1477, Nice, France, October 2003.

-
- [20] Y.G. Jiang, C.W. Ngo, and J. Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. *Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 494–501, 2007.
 - [21] Haibin Ling and Stefano Soatto. Proximity Distribution Kernels for Geometric Context in Category Recognition. In *Proceedings of IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007.
 - [22] J. Hafner, H.S. Sawhney, W. Equitz, M. Flickner, and W. Niblack. Efficient Color Histogram Indexing for Quadratic Form Distance Functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 729–736, 1995.
 - [23] Yan Ke, Rahul Sukthankar, and Larry Huston. Efficient Near-Duplicate Detection and Sub-image Retrieval. In *Proceedings of the 12th annual ACM international conference on Multimedia*, pages 869–876, New York, NY, USA, October 2004.
 - [24] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, New York, NY, USA, June 2006.
 - [25] Kristen Grauman and Trevor Darrell. The Pyramid Matching Kernel: Discriminative Classification with Sets of Image Features. In *Proceedings of IEEE International Conference on Computer Vision*, volume 2, pages 1458–1465, Beijing, China, October 2005.
 - [26] Silvio Savarese, John Winn, and Antonio Criminisi. Discriminative Object Class Models of Appearance and Shape by Correlations. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2033–2040, New York, NY, USA, June 2006.
 - [27] M.R. Naphade, M.M. Yeung, and B.L. Yeo. A novel scheme for fast and efficient video sequence matching using compact signatures. In *Proc. SPIE, Storage and Retrieval for Media Databases*, volume 3972, pages 564–572, 2000.
 - [28] A. Hampapur, K. Hyun, and R. Bolle. Comparison of sequence matching techniques for video copy detection. In *Conf. on Storage and Retrieval for Media Databases*, pages 194–201, 2002.
 - [29] Y. Li, JS Jin, and X. Zhou. Matching commercial clips from TV streams using a unique, robust and compact signature. In *Digital Image Computing: Techniques and Applications, 2005. DICTA'05. Proceedings 2005*, pages 39–39, 2005.

-
- [30] W. Hsu, ST Chua, and HH Pung. An integrated color-spatial approach to content-based image retrieval. In *Proceedings of the third ACM international conference on Multimedia*, pages 305–313. ACM Press New York, NY, USA, 1995.
 - [31] Xiao Wu, Alexander G. Hauptmann, and Chong-Wah Ngo. Practical Elimination of Near-Duplicates from Web Video Search. In *Proceedings of the 15th annual ACM international conference on Multimedia*, pages 218–227, Augsburg, Germany, October 2007.
 - [32] Hung-Khoon Tan, Xiao Wu, Chong-Wah Ngo, and Wan-Lei Zhao. Accelerating Near-Duplicate Video Matching by Combining Visual Similarity and Alignment Distortion. *Proceedings of the 16th annual ACM international conference on Multimedia*, October 2008.
 - [33] M. Weber, M. Welling, and P. Perona. Towards automatic discovery of object categories. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:101–108, 2000.
 - [34] R. Fergus, P. Perona, and A. Zisserman. Weakly Supervised Scale-Invariant Learning of Models for Visual Recognition. *International Journal of Computer Vision*, 71(3):273–303, 2007.
 - [35] L. Fei-Fei, R. Fergus, and P. Perona. One-Shot Learning of Object Categories. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pages 594–611, 2006.
 - [36] P. Felzenszwalb and D Huttenlocher. Pictorial structures for object recognition. *Intl. Journal of Computer Vision*, 61(1):55–79, 2005.
 - [37] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 17–32, 2004.
 - [38] S. Lazebnik, C. Schmid, and J. Ponce. A maximum entropy framework for part-based texture and object recognition. *Proc. ICCV*, 1:832–838, 2005.
 - [39] D.J. Crandall and D.P. Huttenlocher. Composite Models of Objects and Scenes for Category Recognition. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.
 - [40] M. Fischler and R Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 22(1):67–92, 1973.

-
- [41] R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:380–387, 2005.
 - [42] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaf-falitzky, T. Kadir, and L.V. Gool. A Comparison of Affine Region Detectors. *International Journal of Computer Vision*, 65(1):43–72, 2005.
 - [43] S. Tong, D. Koller, and L.P. Kaelbling. Support Vector Machine Active Learning with Applications to Text Classification. *Journal of Machine Learning Research*, 2(1):45–66, 2001.
 - [44] H.M. Wallach. Topic modeling: beyond bag-of-words. *Proceedings of the 23rd international conference on Machine learning*, pages 977–984, 2006.
 - [45] B.C. Russell, A.A. Efros, J. Sivic, W.T. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.
 - [46] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning Object Categories from Google’s Image Search. *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2, 2005.
 - [47] J. Shi and J. Malik. Normalized Cuts and Image Segmentation. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pages 888–905, 2000.
 - [48] K. Mikolajczyk and C. Schmid. Scale & Affine Invariant Interest Point Detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.
 - [49] T. Tuytelaars and L. Van Gool. Matching Widely Separated Views Based on Affine Invariant Regions. *International Journal of Computer Vision*, 59(1):61–85, 2004.
 - [50] T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. *European Conference on Computer Vision*, 1:228–241, 2004.
 - [51] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. *Proc. ECCV*, 4:490–503, 2006.
 - [52] N. Dalai, B. Triggs, I. Rhone-Alps, and F. Montbonnot. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, 2005.

-
- [53] JA Hartigan and MA Wong. A K-means clustering algorithm. *JR Stat. Soc., Ser. C*, 28:100–108, 1979.
- [54] J. Sivic, BC Russell, AA Efros, A. Zisserman, and WT Freeman. Discovering Objects and their Localization in Images. *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 1, 2005.
- [55] A. Böcker, S. Derksen, E. Schmidt, A. Teckentrup, and G. Schneider. A Hierarchical Clustering Approach for Large Compound Libraries. *J. Chem. Inf. Model*, 45(4):807–815, 2005.
- [56] Kristen Grauman and Trevor Darrell. Approximate Correspondences in High Dimensions. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, volume 19, pages 505–512, Cambridge, MA, USA, December 2006.
- [57] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 2161–2168, 2006.
- [58] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
- [59] F. Jurie and B. Triggs. Creating Efficient Codebooks for Visual Recognition. *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 1, 2005.
- [60] D. Comaniciu and P. Meer. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pages 603–619, 2002.
- [61] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*, 19:985, 2007.
- [62] D. Larlus and F. Jurie. Latent mixture vocabularies for object categorization. *British Machine Vision Conference*, 2006.
- [63] D.M. Blei, A.Y. Ng, M.I. Jordan, and J. Lafferty. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3(4-5):993–1022, 2003.
- [64] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. *Proc. ICCV*, 83:84–86, 2005.

-
- [65] F. Perronnin, C. Dance, G. Csurka, and M. Bressan. Adapted vocabularies for generic visual categorization. *European Conference on Computer Vision*, 3954:464–475, 2006.
 - [66] S. Lazebnik, C. Schmid, and J. Ponce. Semi-local affine parts for object recognition. *British Machine Vision Conference*, 2:959–968, 2004.
 - [67] F. Jurie, C. Schmid, I.C. Gravir, and F. Montbonnot. Scale-invariant shape features for recognition of object categories. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2, 2004.
 - [68] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. *European Conference on Computer Vision*, pages:304–317, 2008.
 - [69] M. Perd'oc, O. Chum, and J. Matas. Efficient representation of local geometry for large scale object retrieval. *IEEE Conference on Computer Vision and Pattern Recognition*, 6(37146):9–16, 2009.
 - [70] S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighborhood statistics for texture recognition. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 649–655, 2003.
 - [71] J. Blanchet, F. Forbes, and C. Schmid. Markov Random Fields for Recognizing Textures modeled by Feature Vectors. In *International Conference on Applied Stochastic Models and Data Analysis, France*, 2005.
 - [72] J. Blanchet, F. Forbes, C. Schmid, T. Lear, and I.R.A. Mistis. Markov random fields for textures recognition with local invariant regions and their geometric relationships. In *British Machine Vision Conference*.
 - [73] J. Lafferty, A. McCallum, and F. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*, pages 282–289, 2001.
 - [74] J. Li, W. Wu, T. Wang, and Y. Zhang. One step beyond histograms: Image representation using Markov stationary features. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Anchorage, Alaska*, 2008.
 - [75] B. Ni, S. Yan, and A. Kassim. Contextualizing histogram. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009.

-
- [76] M. Marszałek and C. Schmid. Spatial Weighting for Bag-of-Features, June 2006.
 - [77] Serge Belongie, Jitendra Malik, and Jan Puzicha. Distinctive Image Features from Scale-Invariant Keypoints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002.
 - [78] TREC Video Retrieval Evaluation (TRECVID). <http://www.nlpir.nist.gov/projects/trecvid/>.
 - [79] Junsong Yuan, Ying Wu, and Ming Yang. Discovery of Collocation Patterns: from Visual Words to Visual Phrases. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, June 2007.
 - [80] Liangliang Cao and Li Fei-Fei. Spatially Coherent Latent Topic Model for Concurrent Object Segmentation and Classification. In *Proceedings of IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007.
 - [81] David Nistér and Henrik Stewenius. Scalable Recognition with a Vocabulary Tree. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2161–2168, New York, NY, June 2006.
 - [82] Dong Xu and Shih-Fu Chang. Visual Event Recognition in News Video using Kernel Methods with Multi-Level Temporal Alignment. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, June 2007.
 - [83] Yan-tao Zheng, Ming Zhao, Shi-Yong Neo, Tat-Seng Chua, and Qi Tian. Visual synset: Towards a higher-level visual representation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Anchorage, Alaska*, pages 1–8, 2008.
 - [84] Zhong Wu, Qifa Ke, Michael Isard, and Jian Sun. Bundling features for large scale partial-duplicate web image search. *IEEE Conference on Computer Vision and Pattern Recognition (2009)*, 09:25–32, 2009.
 - [85] Jianke Zhu, Steven C.H. Hoi, Michael R. Lyu, and Shuicheng Yan. Near-Duplicate Keyframe Retrieval by Nonrigid Image Matching. *Proceedings of the 16th annual ACM international conference on Multimedia*, October 2008.
 - [86] Y. Boykov and D.P. Huttenlocher. A new bayesian framework for object recognition. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 2002.

-
- [87] M.A. Abdulrahim and M. Misra. A graph isomorphism algorithm for object recognition. *Pattern Analysis & Applications*, 1(3):189–201, 1998.
 - [88] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.
 - [89] D. Xu, T.J. Cham, S. Yan, and S.F. Chang. Near Duplicate Image Identification with Spatially Aligned Pyramid Matching. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7, 2008.
 - [90] Xiao Wu, Wan-Lei Zhao, and Chong-Wah Ngo. Near-Duplicate Keyframe Retrieval with Visual Keywords and Semantic Context. In *Proceedings of International Conference on Image and Video Retrieval (CIVR)*, pages 162–169, Amsterdam, The Netherlands, July 2007.
 - [91] W. Zhao, Y. Jiang, and C. Ngo. Keyframe Retrieval by Keypoints: Can Point-to-Point Matching Help? *Proceedings of International Conference on Image and Video Retrieval*, 2006.
 - [92] J. Han and K.K. Ma. Fuzzy color histogram and its use in color image retrieval. *Image Processing, IEEE Transactions on*, 11(8):944–952, 2002.
 - [93] M.J. Pickering, D. Heesch, S. Ruger, R. O Callaghan, and D. Bull. Video retrieval using global features in keyframes. *NIST SPECIAL PUBLICATION SP*, pages 318–324, 2003.
 - [94] A. Qamra, Y. Meng, and E.Y. Chang. Enhanced Perceptual Distance Functions and Indexing for Image Replica Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 379–391, 2005.
 - [95] H.J. Zhang, J. Wu, D. Zhong, and S.W. Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern recognition*, 30(4):643–658, 1997.
 - [96] M. Datar, N. Immorlica, P. Indyk, and V.S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, pages 253–262. ACM, 2004.
 - [97] G. Ahanger and T.D.C. Little. A Survey of Technologies for Parsing and Indexing Digital Video. *Journal of Visual Communication and Image Representation*, 7(1):28–43, 1996.
 - [98] T.G.A. Smith and N.C. Pincever. Parsing movies in context. In *Proc Summer 1991 Usenix Conference*, pages 157–168.

- [99] T.S. Chua, L. Chen, and J. Wang. Stratification approach to modeling video. *Multimedia Tools and Applications*, 16(1):79–97, 2002.
- [100] Y. Rubner, C. Tomasi, and L.J. Guibas. The Earth Mover’s Distance as a Metric for Image Retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.
- [101] T. Serre, M. Riesenhuber, J. Louie, and T. Poggio. On the role of object-specific features for real world object recognition in biological vision. In *Biologically Motivated Computer Vision*, pages 387–397, 2002.
- [102] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, June 2005.
- [103] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2009.
- [104] H.W. Kuhn. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, pages 83–97, 1955.
- [105] C.C. Chang and C.J. Lin. LIBSVM: a library for support vector machines, 2001. 2001.
- [106] L.J. Li and L. Fei-Fei. What, where and who? classifying event by scene and object recognition. In *IEEE International Conference on Computer Vision*, October 2007.
- [107] J. Wu and J.M. Rehg. Beyond the Euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. *Proceedings of IEEE International Conference on Computer Vision*, 2009.
- [108] Jianxin Wu and James M. Rehg. Where am i: Place instance and category recognition using spatial pact. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Anchorage, Alaska*, 2008.