Advanced Techniques for Semantic Concept Detection in General Videos

Wei Jiang

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2010

ABSTRACT

Advanced Techniques for Semantic Concept Detection in General Videos

Wei Jiang

The automatic analysis and indexing of multimedia content in general domains are important for a variety of multimedia applications. This thesis investigates the problem of semantic concept detection in general videos focusing on two advanced directions: multi-concept learning and multi-modality learning.

Semantic concept detection refers to the task of assigning an input video sequence one or multiple labels indicating the presence of one or multiple semantic concepts in the video sequence. Much of the prior research work deals with the problem in an isolated manner, *i.e.*, a binary classifier is constructed using feature vectors from the single visual modality to classify whether or not a video contains a specific concept. However, multimedia videos comprise of information from multiple modalities (both visual and audio). Each modality brings some information about the other and their simultaneous processing can uncover relationships that are otherwise unavailable when considering the modalities separately. In addition, real-world semantic concepts do not occur in isolation. The context information is useful for enhancing detection of individual concepts.

This thesis explores multi-concept learning and multi-modality learning to improve semantic concept detection in general videos, *i.e.*, videos with general content and are captured in uncontrolled conditions. For multi-concept learning, we propose two methods with the frameworks of two-layer Context-Based Concept Fusion (CBCF) and single-layer multi-label classification, respectively. The first method represents the inter-conceptual relationships by a Conditional Random Field (CRF). The inputs of the CRF are initial detection probabilities from independent concept detectors. Through inference with concept relations in the CRF we get updated concept detection probabilities as outputs. To avoid the difficulty

of designing compatibility potentials in the CRF, a discriminative cost function aiming at class separation is directly minimized. Also, we further extend this method to study an interesting "20 questions problem" for semantic concept detection, where user's interaction is incorporated to annotate a small number of key concepts for each data, which are then used to improve detection of the remaining concepts. To this end, an active CBCF approach is proposed that can choose the most informative concepts for the user to label. The second multi-concept learning method does not explicitly model concept relations but optimizes multi-label discrimination for all concepts over all training data through a single-layer joint boosting algorithm. By sharing "good" kernels among different concepts, accuracy of individual detectors can be improved; by joint learning of common detectors across different classes, required kernels and computational complexity for detecting individual concepts can be reduced.

For multi-modality learning, we develop methods with two strategies: global fusion of features or models from multiple modalities, and construction of the local audio-visual atomic representation to enforce a moderate-level audio-visual synchronization. Two algorithms are developed for global multi-modality fusion, *i.e.*, the late-fusion audio-visual boosted CRF and the early-fusion audio-visual joint boosting. The first method is an extension of the above two-layer CBCF multi-concept learning approach where the inputs of the CRF include independent concept detection probabilities obtained by using both visual and audio features, individually. The second method is an extension of the above single-layer multi-label classification approach, where both visual-based kernels and audio-based kernels are shared by multiple concepts through the joint boosting multi-label concept detector. These two proposed methods naturally combines multi-modality learning and multi-concept learning to exert the power of both for enhancing semantic concept detection. To analyze moderate-level audio-visual synchronization in general videos, we propose to generate a local audio-visual atomic representation, *i.e.*, the Audio-Visual Atom (AVA). We track visually consistent regions in the video sequence to generate visual atoms. At the same time we locate audio onsets in the audio soundtrack to generate audio atoms. Then visual atoms and audio atoms are combined together to form AVAs, on top of which joint audio-visual codebooks are constructed. The audio-visual codebooks capture the co-occurring audio-visual patterns that are representative to describe different individual concepts, and accordingly can improve concept detection.

The contributions of this thesis can be summarized as follows. (1) An in-depth study of jointly detecting multiple concepts in general domains, where concept relationships are hard to compute. (2) The first system to explore the "20 questions" problem for semantic concept detection, by incorporating users' interactions and taking into account joint detection of multiple concepts. (3) An in-depth investigation of combining audio and visual information to enhance detecting generic concepts. (4) The first system to explore the localized joint audio-visual atomic representation for concept detection, under challenging conditions in general domains.

Table of Contents

1 Introduction 1

1.1 Background 1

1.2 Our Approach 3

1.2.1 Multi-Concept Learning 4

1.2.2 Multi-Modality Learning 9

1.3 Unique Contributions 12

1.4 Thesis Overview 13

2 Review of Concept Detection in Images and Videos 15

2.1 Terminology 15

2.2 Related Works on Image Classification 17

2.2.1 Visual Descriptors 17

2.2.2 Kernel-based Classification 20

2.2.3 Evaluation 22

2.3 Related Works on Object/Region Recognition in Images 23

2.3.1 Object Localization 24

2.3.2 Evaluation of Object Localization 25

2.3.3 Multi-Class Image Segmentation 26

2.4 Related Works on Classifying Dynamic Video Sequences 27

2.4.1 Object-Level Action Recognition 28

2.4.2 Evaluation of Action Recognition 30

2.4.3 Generic Concept Detection by Classifying Video Sequences 31

2.5 Related Works on Multi-Modality Learning 31
2.5.1 Audio-Visual Speech and Speaker Recognition 32
2.5.2 Audio-Visual Synchronization for Localization 32
2.5.3 Multi-Modal Fusion for Video Classification 33
2.6 Related Works on Multi-Concept Learning 34
2.6.1 Object and Region-based Recognition with Context Information . . 35
2.6.2 Image and Video Classification with Context Information 37
2.7 Data Sets and Performance Measures 38
2.7.1 Construction of Kodak's Benchmark Consumer Video Set 39
2.7.2 TRECVID 2005 Data Set 45
2.7.3 Performance Evaluation Metric 46
2.7.4 Global Visual Features and Baseline Models 46
2.7.5 BoF over Kodak's Consumer Benchmark Videos 47

3 Multi-Concept: Context-Based Concept Fusion 55

3.1 Introduction 55
3.2 Boosted CRF Concept Fusion 58
3.2.1 Boosted CRF 59
3.2.2 Boosted CRF–Concept Fusion 60
3.3 Which Concepts to Update 61
3.4 Experiments: BCRF-CF 63
3.4.1 Performance Evaluation 64
3.4.2 Evaluation with Different Parameters 65
3.5 20 Questions Problem: Active CBCF 67
3.5.1 The Active CBCF approach 68
3.5.2 Maximizing Mutual Information 69
3.5.3 Further Consideration 70
3.6 Experiments: Active CBCF 71
3.7 Unique Contributions 73

4 Multi-Concept: Sharing Kernels and Classifiers 75

4.1 Introduction 75

4.2 Construction of Vocabulary/Kernel Pool 78

4.2.1 Vocabulary Construction 78

4.2.2 Kernel Construction 79

4.3 Joint Learning with Joint Boosting 80

4.3.1 Learning with Boosting 81

4.3.2 Joint Boosting and Kernel Sharing 83

4.4 Experiments 85

4.4.1 Comparison between Different Kernels 87

4.4.2 Vocabulary Sharing with Joint Boosting 88

4.4.3 Exploring Concept Relations 90

4.4.4 Computational Complexity 92

4.5 Discussion 94

5 Multi-Modal Concept Detection by Direct Fusion 96

5.1 Introduction 96

5.2 Audio-based Concept Detector 98

5.2.1 Single Gaussian Modeling 99

5.2.2 Probabilistic Latent Semantic Analysis 99

5.3 Audio-Visual Boosted CRF 101

5.4 Audio-Visual Joint Boosting 102

5.5 Experiments 102

5.6 Discussion 107

6 Multi-Modality: Mid-Level A-V Atoms 109

6.1 Introduction 109

6.2 Overview of Our Approach 112

6.3 Visual Atom Extraction 114

6.3.1 Short-Term Point Track 116

6.3.2 Short-Term Region Track 116

6.3.3 Visual Features over Region Tracks 119

6.3.4 Long-term Linking to Generate Visual Atoms 121

6.3.5 Refinement through Clustering 123

6.4 Audio Atom Extraction 124

6.4.1 Extracting Time-Frequency Bases 124

6.4.2 Locate Audio Onsets 126

6.4.3 Audio Atoms and Features 127

6.4.4 Refinement through Clustering 127

6.5 Joint Audio-Visual Codebook Construction 128

6.6 Classification with Joint A-V Codebooks 131

6.7 Experiments 131

6.7.1 Codebook Visualization 133

6.7.2 Performance of Concept Detection 138

6.8 Discussion 142

7 Conclusions **145**

7.1 Summary 145

7.1.1 Multi-Concept Learning 145

7.1.2 Multi-Modality Learning 146

7.2 Future Work 147

7.2.1 Multi-Concept Learning 148

7.2.2 Multi-Modality Fusion 148

7.2.3 Mid-Level Audio-Visual Correlation 149

7.2.4 Fusion of Multi-Concept & Multi-Modality Learning 150

Bibliography **151**

List of Figures

1.1 The framework of a semantic concept detection system. 2

1.2 The three strategies of multi-concept learning using context-based concept fusion. 6

1.3 Sharing features and classifiers to help detect individual concepts. 8

2.1 Data structure of a video concept detection task. 16

2.2 Example keyframes for Kodak's consumer benchmark video set corresponding to different semantic concepts. 45

2.3 Four spatial partitions of images to generate BoF histograms. 50

2.4 Performance comparison: global features versus local SIFT over different interest keypoints. 51

2.5 Performance comparison: global features versus different SIFT-based descriptors over the Harris-Laplace detector. 52

2.6 Performance comparison: global features versus local descriptors. 53

2.7 Example of Kodak's consumer videos illustrating the challenging conditions for concept detection. 54

3.1 The CRF that models relationships among M concepts. 56

3.2 The BCRF-CF algorithm. 62

3.3 The MAP of BCRF-CF, DMF, and independent concept detectors. The results are averaged over 26 selected concepts. 65

3.4 The individual AP comparison of our BCRF-CF, DMF, and the baseline independent detector. 66

3.5 Examples of performance evolution. 67

3.6 The work flow of Active CBCF. 69

3.7 Performance with 1 concept labeled for each keyframe. 73

3.8 MAP with different numbers of labeled concepts. 74

3.9 Example images where Active CBCF can effectively select the most informative concepts for the user to label. 74

4.1 Examples of vocabulary/kernel construction. 77

4.2 Illustration of learning procedures. 82

4.3 The joint boosting algorithm based on Real AdaBoost. 86

4.4 The list of 12 concepts and associated example keyframes for evaluating the proposed kernel/classifier sharing method. 87

4.5 Per-concept AP: VSPM baseline vs. single-vocabulary kernels. l indicates the vocabulary resolution level. 89

4.6 Per-concept AP of joint boosting and two baseline approaches: VSPM baseline and regular boosting. 90

4.7 Performance evolution of joint boosting and regular boosting with different numbers of weak classifiers. 91

4.8 Usage of individual kernels by 100 weak classifiers. 92

4.9 Sharing of kernels among different concepts. Each bar gives the frequency a kernel is used in building weak classifiers over 100 iterations for a concept. . 93

4.10 Evolution of AP performance through iterations. 94

5.1 Early fusion and late fusion of audio and visual modalities for concept detection. 97

5.2 Comparison of different audio-visual fusion algorithms. 104

5.3 Top 20 video segments from the “parade” concept. 105

5.4 Frequency of kernels used by the A-V Joint Boosting algorithm throughout 200 iterations. 107

6.1 Examples of audio-visual atoms. 110

6.2 The overall framework of the proposed joint audio-visual analysis approach. 112

6.3 An example of region track replication. 117

6.4 The algorithm to generate short-term region tracks. 118

6.5 Numbers of point tracking trajectories with changing lengths of the short-term video slice. 119

6.6 Examples of fast moving objects that can only be captured when we track visual regions at 30 fps within 0.3-sec video slices. 120

6.7 The algorithm for long-term linking to generate visual atoms. 122

6.8 Examples of the final visual atoms extracted from a "wedding" video. . . . 123

6.9 Illustration of incomplete visual tracking. The visual atoms may only capture a consistent region over discontinued time windows. 124

6.10 Examples of clustered visual atoms. 125

6.11 An example of the original audio signal and the corresponding extracted time-frequency bases. 126

6.12 Example of clustered audio atoms. 128

6.13 Structure of the AVAs in our implementation. This structure can be easily extended to accommodate other types of features. 129

6.14 The algorithm to construct concept detectors by selectively using different codebooks. 132

6.15 Example prototypes learned with audio-visual atoms for concepts of several broad categories, *i.e.*, Locations, Activities, Scenes, Occasions, and Objects. 135

6.16 Examples of learned static region prototypes by DD-SVM. 136

6.17 Example prototypes learned with audio-visual atoms for concepts that are expected to have strong cues in both audio and visual aspects. 137

6.18 Comparison of visual atoms with static-region-based methods. 139

6.19 Comparison of AVA-MIL with individual $\mathbf{f}_{vis}$, $\mathbf{f}_{loc}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$. . . . 140

6.20 Comparison of AVA-MIL with different combinations of $\mathbf{f}_{vis}$, $\mathbf{f}_{loc}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$. 141

6.21 Comparison of multi-modal AVA-MIL-Boosting, VA-MIL with $\mathbf{f}_{vis}$ alone, and direct fusion. 143

List of Tables

2.1 Selected concepts and definitions. 42

2.2 Information of Kodak's video data set. 43

2.3 The number of positive and negative keyframes and video clips in Kodak's consumer video data set. 44

3.1 The data sets for experiments to evaluate BCRF-CF. 64

3.2 Performance of detection with different β. 66

3.3 The data sets for experiments to evaluate Active CBCF. 71

4.1 MAP: VSPM baseline vs. single-vocabulary kernels (with varying numbers of levels used in the spatial pyramid). 88

Acknowledgments

I owe my gratitude to all those people who have made this thesis possible. It is a pleasure to thank them all in my humble acknowledgment.

In the first place, I am heartily thankful to my supervisor, Professor Shih-Fu Chang, for his supervision, guidance, encouragement, and support throughout my Ph.D study. His scientific intuition and rigorous research attitude have enriched me with the in-depth understanding of the subject and the passion of dedicating myself to research, which I will benefit from in the long run. His gracious support in many aspects has enabled me to overcome difficulties and finish my study.

This is a great opportunity to express my respect and gratitude to Dr. Alexander Loui for his supervision, advice, and insightful comments at different stages of my research. I am thankful to him for being a wonderful mentor and a great source of support and encouragement.

Many thanks go in particular to Professor Dan Ellis, for his valuable advice in science discussion and his crucial contribution to our research projects.

I am also grateful to Professor Tony Jebara and Shree Nayar for the lectures on related topics that have greatly improved my knowledge in the area.

I would like to thank Dr. Lexing Xie for the numerous discussions and her encouragements. I truly appreciate her help in many ways.

I am also thankful to the following former or current colleagues in the DVMM lab and LabROSA of Columbia University: Courtenay Cotton, Junfeng He, Yu-Feng Hsu, Winston Hsu, Yu-Gang Jiang, Keansub Lee, Wei Liu, Tian Tsong Ng, Jun Wang, Akira Yanagawa and Eric Zavesky. Because of them my graduate study has become an enjoyable memory that I will cherish forever.

I would like to convey my gratitude to the Kodak Research Labs in Eastman Kodak

Company for the fruitful interactions and collaborations, especially for funding my Ph.D study. Also, I would like to thank the multimedia research teams in IBM T. J. Watson Research center for their collaborations.

I am heartily grateful to my parents for their constant love and support throughout all these years. None of this would have been possible without them.

Finally, I would like to express my special gratitude to my husband, Jinwei Gu, whose dedication, love, and persistent confidence in me has diminished all the difficulties of studying and living in the unfamiliar land far away from home.

Glossary

Active CBCF – Active Context-Based Concept Fusion

AP – Average Precision

AVA – Audio-Visual Atom

A-V Boosted CRF – Audio-Visual Boosted Conditional Random Field

A-V Joint Boosting – Audio-Visual Joint Boosting

BCRF-CF – Boosted Conditional Random Field - Concept Fusion

BoF – Bag of Features

CBCF – Context-Based Concept Fusion

CRF – Conditional Random Field

DD – Diverse Density

DMF – Discriminative Model Fusion

DoG – Difference-of-Gaussian

EDH – Edge Direction Histogram

GBR – Gabor texture

GCM – Grid Color Moments

GMM – Gaussian Mixture Model

HMM – Hidden Markov Model

HOG – Histogram of Oriented Gradient

KL – Kullback-Leibler

KLT – Kanade-Lucas-Tomasi

LSCOM – Large-Scale Concept Ontology for Multimedia

MAP – Mean Average Precision

MFCC – Mel-Frequency Cepstral Coefficient

MIL – Multiple Instance Learning

MP – Matching Pursuit

MRF – Markov Random Field

MSER – Maximally Stable Extreme Regions

pLSA – probabilistic Latent Semantic Analysis

SIFT – Scale Invariant Feature Transform

SPM – Spatial Pyramid Matching

STFT – Short-Time Fourier Transform

STR-PTRS – Short-Term Region tracking with joint Point Tracking and Region Segmentation

SVM – Support Vector Machine

VSPM – Vocabulary-Spatial Pyramid Matching

Chapter 1

Introduction

In this thesis, we propose several new methods to detect generic semantic concepts in general videos, by exploiting information from multiple modalities, and by jointly considering multiple concepts.

1.1 Background

With the prevalent use of digital video capturing devices and online search engines, most users are now accustomed to simple and intuitive interfaces when interacting with large video sources. Since multimedia videos comprise of unstructured information that usually is not described by associated text keywords, *semantic concept detection* (also called *semantic-level indexing* or *high-level feature extraction* in some literatures) is needed to extract high-level information from raw video data. Semantic concept detection is defined as the task of assigning an input video one or multiple labels indicating the presence of one or multiple *semantic concepts* in the video sequence. Such semantic concepts can be anything of users' interest that are visually observable, such as objects (*e.g.*, "car" and "people"), activities (*e.g.*, "running" and "laughing"), scenes (*e.g.*, "sunset" and "beach"), events (*e.g.*, "birthday" and "graduation"), *etc.* Semantic concept detection systems enable automatic indexing and organization of massive multimedia information, which provide important supports for a broad range of applications such as content or keyword-based multimedia search, content-based video summarization, robotic vision, and so on.

Development of large-scale semantic concept detection systems requires several critical components. First, an ontology is needed to define a list of important concepts and the relations among the concepts. Such ontologies may be constructed based on formal user studies or data mining of user interaction logs with online systems. Second, a large corpus consisting of realistic data is needed for training and evaluation. An annotation process is also needed to obtain the labels of the defined concepts over the corpus. Third, multimedia processing and machine learning tools are needed to develop robust *classifiers* (also called *models* or *concept detectors*) that can be used to detect the presence of each concept in a video. Fig. 1.1 describes the framework of a typical semantic concept detection system.

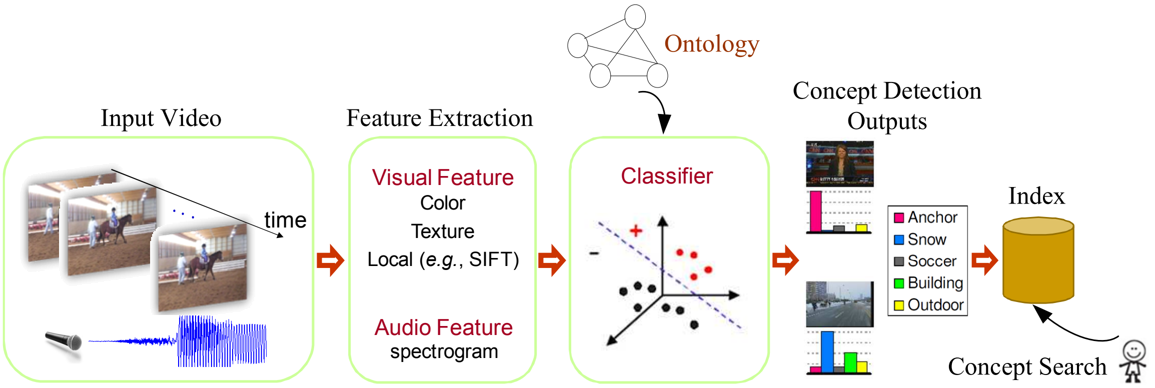


Figure 1.1: The framework of a semantic concept detection system.

As will be shown in detail in Chapter 2, researchers have developed many concept detection systems for various applications. These approaches can be categorized into two broad categories: one for detecting concepts at the object or region level, the other detecting concepts in the whole images/videos. In this thesis, we focus on methods that belong to the second category, due to their compatibility and flexibility in handling a broad range of concepts and videos. Methods in the first category detect and locate either individual objects/regions (*e.g.*, “car”, “boat”, “people”, and “sky”), actions of objects (*e.g.*, “people walking” and “arm lifting”), or dynamic events that comprise of multiple objects and their interactions in the video (*e.g.*, “car exiting the parking lot”, and “a person leaving his/her baggage”). Although accurate detection of objects, regions, their actions and interactions are useful towards the total understanding of images and videos, the tasks are very challenging. This is because of the difficulties in determining the locations and scales of target objects/regions, the variability produced by scene conditions like illumination, background, clutter and occlusion, and the difficulty in obtaining the object-level label annotation. As a result, detection in general videos, *i.e.*, videos with general content and are captured in uncontrolled conditions, is still far from satisfactory. Object-level approaches usually work only in controlled videos or surveillance applications with fixed (or almost fixed) cameras and stable backgrounds. For the purpose of detecting generic concepts in general videos, the global detection methods are relatively more realistic and thus adopted as the focus of this thesis.

1.2 Our Approach

Traditional concept detection approaches classify images and videos in an isolated manner. That is, to classify whether an image or video contains a specific concept, a binary classifier is trained using some feature vectors from a single modality (*e.g.*, visual data alone) and some statistic recognition models so that a probabilistic judgement can be made for a test image or video.

Videos comprise of information from multiple modalities, both visual and audio. Each modality brings information complementary with the others and simultaneous processing of such sources can uncover relationships that are otherwise unavailable. The need of integrating multi-model information is confirmed in human perception systems also [3, 45, 69]. As a result, audio-visual multimedia data analysis has received more interest in recent years [7, 33, 89, 113, 142].

In addition, in the real world semantic concepts usually do not occur in isolation. Such context information has been shown important for enhancing concept detection accuracy [102, 162, 192, 198, 238]. For example, a confident detection of one concept (*e.g.*, "car") may provide a strong cue about the high likelihood of detecting another concept (*e.g.*, "road") [61, 112]. In this thesis, we will develop several novel methods for improving the performance of detecting generic concepts in general videos, by focusing on the approaches of multi-concept learning and multi-modality learning.

1.2.1 Multi-Concept Learning

In previous literatures (see Section 2.6 for a detailed review), there are different levels of contexts that have been used for multimedia content analysis. For example, context can model the local smoothness of pixel labels, the spatial relationship among image regions and objects (*e.g.*, a boat is usually on top of water), and the co-occurrence of objects (*e.g.*, a computer usually co-occurs with a keyboard). Such context information has been successfully used in object-level and region-level detection [61, 112, 209, 222, 224]. In addition, context can model the relationship (*e.g.*, causal, co-occurrence, *etc.*) among semantic concepts. Since we focus on detection of concepts from the whole image or video instead of object localization, the context describing relationships among semantic concepts is the main focus of our work in this thesis.

1.2.1.1 Two-Layer Context-Based Concept Fusion

Recently, a *Context-Based Concept Fusion* (*CBCF*) framework has been proposed (first by Naphade *et al.* in [162]) for semantic concept detection in general videos. CBCF has a two-layer learning structure. In the first layer, independent concept detectors are applied to get posterior probabilities of individual concepts of a given image or video. Then in the second layer detection results of individual concepts are updated through a context-based model by treating detection confidence scores of all concepts as contexts. Earlier CBCF approaches use pre-defined concept relations such as the ontology hierarchy [246] and the Bayesian networks (manually constructed in most cases) [179], which can only be used in specific domains with prior knowledge about the structure of ontology. On the other hand, the *vector model* approach does not explicitly model concept relationships [214]. Instead, initial concept detection scores are used as representation features, on top of which discriminative classifiers such as *Support Vector Machines* (*SVMs*) [230] are constructed to model the proximity relation among concepts. In recent years, the graph-based approach has become popular, where graphical models are used to encode the context information, such as [162] and [102], with nodes representing concepts, the single-node potentials corresponding to concept detection scores, and the two-node compatibility potentials reflecting the pairwise conceptual relations, *e.g.*, the co-occurrence statistical relationships of concepts. The concept detection results are then refined through graph inference.

CBCF, especially the graph-based approach, provides an intuitive framework for incorporating inter-conceptual relationships with significant flexibility, *e.g.*, to adopt various classifiers as single-node concept detectors and to incorporate different types of prior knowledge in building graph structures. The main issue of graph-based CBCF methods, however, lies in the difficulty of designing appropriate compatibility potentials. Previous methods use co-occurrence statistics of concepts to approximate pairwise concept relations. Such approximations become unreliable when we only have limited training data. It is difficult to obtain accurate co-occurrence statistics involving diverse generic concepts in general videos. In this case, the errors from the inaccurate approximation of concept correlation will easily amount to major performance drop.

In this thesis, we develop a multi-concept learning algorithm, namely *Boosted CRF–Concept Fusion* (*BCRF-CF*), under the framework of CBCF [97]. We model the inter-conceptual relationships by a *Conditional Random Field* (*CRF*) [114], which models the conditional distribution of concept labels given the initial detection probabilities from independent concept detectors. To avoid the difficulty of designing compatibility potentials in CRF, a discriminative cost function aiming at class separation is directly minimized. That is, we do not use the co-occurrence statistics of concepts to estimate compatibility potentials. Instead, we transfer the problem of learning compatibility potentials into modeling the proximity relation among concepts, by learning a discriminative classifier using initial concept detection scores from independent detectors. The idea is similar to the vector model approach with the difference that we enforce a CRF graph structure to encode pairwise concept relations. Thus our BCRF-CF inherits the intuitiveness and flexibility of graph-based CBCF in incorporating inter-conceptual relationships while avoiding designing compatibility potentials explicitly. Fig. 1.2 illustrates the three multi-concept learning strategies in CBCF. Specifically we minimize a cost function derived from the CRF targeting at discrimination for all concepts over all training data. Additionally, the Real AdaBoost algorithm [59] is incorporated to iteratively refine concept detection performance. We propose a simple but effective criterion to predict which concepts will benefit from CBCF, based on both information theoretic and heuristic rules. This criterion takes into consideration both the strength of relationships between a concept and its neighborhood and the robustness of detections of this neighborhood. Such prediction scheme allows us to use CBCF wisely, applying it only when it is likely to be effective.

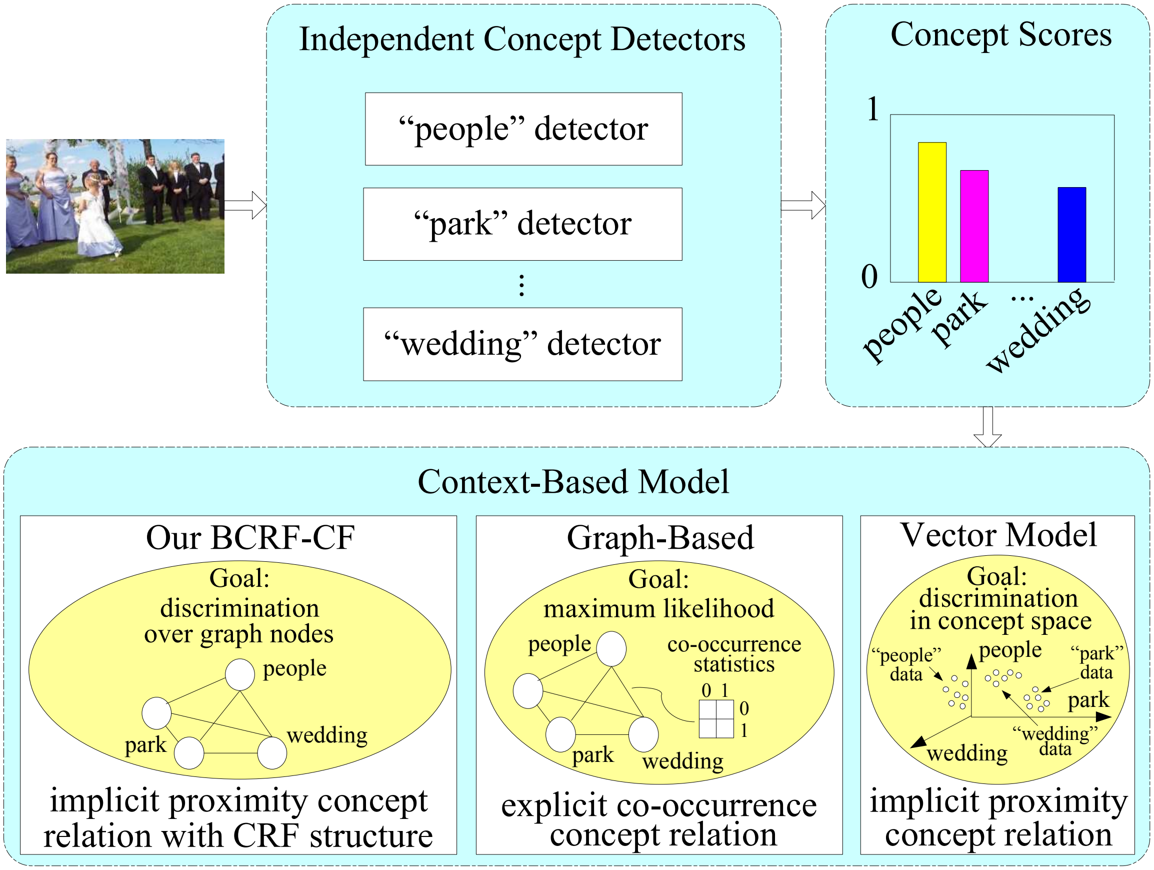


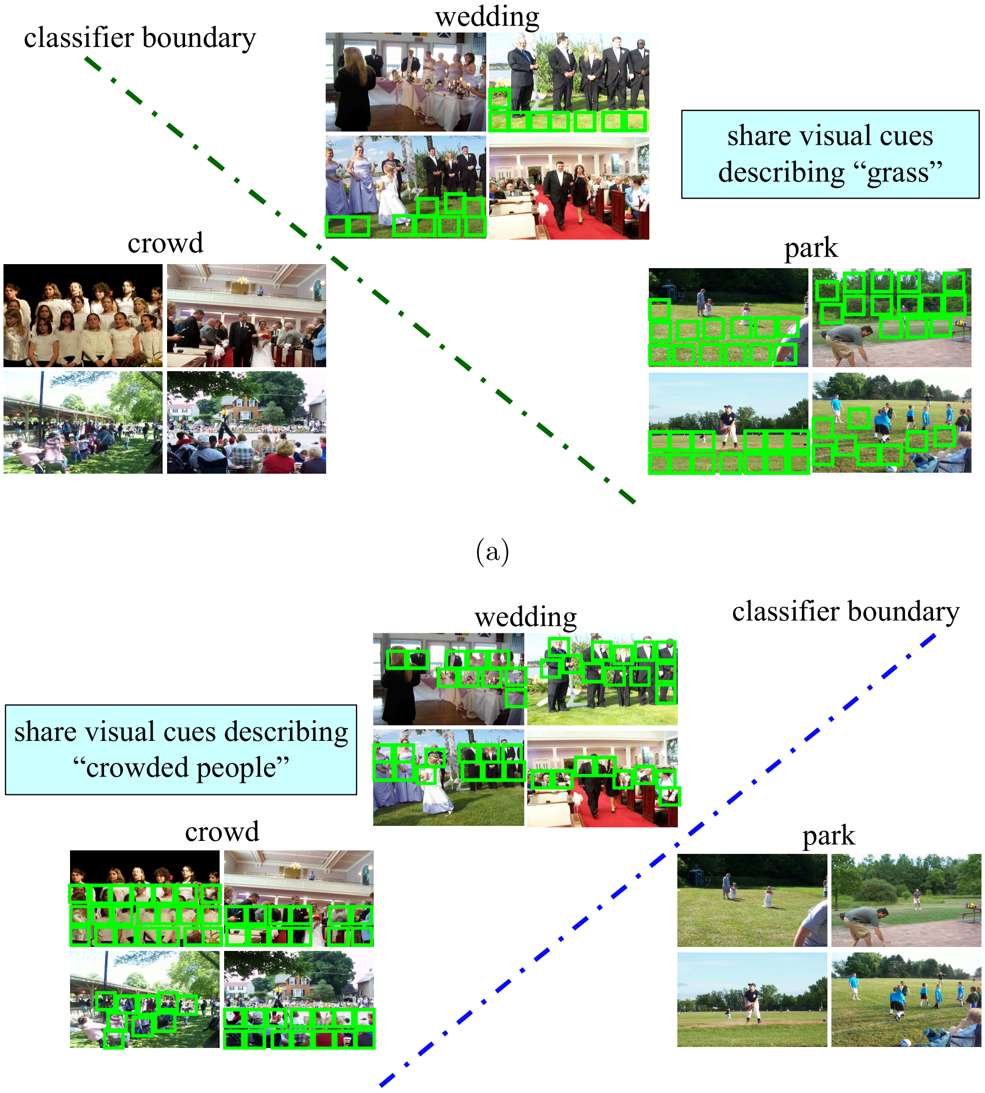
Figure 1.2: The three strategies of multi-concept learning using context-based concept fusion.

Based on the CBCF framework, we further study an interesting active learning scenario: the "20 questions problem" in semantic concept detection [96]. In this scenario, user's input is used to annotate a small subset of concepts per image, which are then used to improve detection of a large number of remaining concepts. To this end, we propose a new paradigm, called *Active Context-Based Concept Fusion* (*Active CBCF*), to adaptively select the right concepts, different for each image, for user annotation. This is in contrast to conventional passive methods where users are asked to annotate all concepts or a subset of arbitrarily chosen concepts.

1.2.1.2 Single-Layer Multi-Label Classification

Although CBCF provides a flexible and intuitive framework to incorporate inter-conceptual relationships, the two-layer framework may suffer from the following potential problems: detection errors of independent detectors in the first layer can accumulate through learning of the context-based model in the second layer; and we need to split the training set into two subsets for learning independent detectors and context-based fusion model, respectively. To address these issues, Qi *et al.* [192] have developed a correlative multi-label method where concept relations are modeled by a Gibbs random field, and the graph structure and concept detectors are learned together in a single step. However, due to the high complexity (at least quadratic to the number of concepts), this method becomes infeasible for practical applications when the number of concepts is large (*e.g.*, dozens or hundreds).

Following the idea of using a single step to learn multi-concept detectors, we also develop a multi-label classification method based on kernel sharing and joint learning [98]. Like the approach of BCRF-CF, we do not explicitly model concept relations but optimize multi-label discrimination for all concepts over all training data. The motivation is clear: by sharing visual cues or similarity kernels from multiple related concepts, each individual concept can be enhanced by adding the descriptive power from others. For instance, as illustrated in Fig. 1.3, "wedding" is usually hard to detect due to the diverse object appearances and the lacking of strong visual cues. Reliable visual cues may be obtained from the data subset labeled with "park" to describe the green grass, which can be shared by the data subset labeled with "wedding" and can be used to separate the combined "wedding" and "park" set from data of the other concepts. Similarly, the reliable visual cues extracted from the "crowd" class can be shared by the "wedding" class and can be used to separate the combined "wedding" and "crowd" set from others. With the help of sharing common classifiers and kernels among different concepts, the "wedding" concept can be better classified by the multi-label concept detector. By sharing "good" kernels among different concepts, accuracy of individual detectors can be improved; by joint learning of common detectors across different classes, the overall computational complexity for detecting concepts can be reduced.


classifier boundary
wedding
share visual cues describing “grass”
crowd
park
wedding
classifier boundary
share visual cues describing “crowded people”
crowd
park

(a)

(b)

Figure 1.3: Sharing features and classifiers to help detect individual concepts. (a) Visual cues (describing grass) obtained from the “park” data are shared by the “wedding” data and are used to separate “wedding” and “park” from other concepts. (b) Visual cues (describing people) obtained from the “crowd” data are shared by the “wedding” data and are used to separate “wedding” and “crowd” from other concepts.

Specifically, in this approach we propose a new kernel construction algorithm, *Vocabulary-Spatial Pyramid Matching* (*VSPM*), which constructs multi-resolution visual vocabularies by hierarchical clustering of local visual features over interest points and computes similarity kernels based on spatial matching. VSPM combines the power of multi-resolution spatial matching and multi-layer vocabulary fusion. Such a combined method is used to generate multiple vocabularies/kernels for each concept, which are further aggregated and shared by all concepts. To jointly learn kernel-based detectors and to share kernels from different concepts, we propose a joint boosting algorithm to automatically select the optimal kernels and the subsets of sharing concepts in an iterative boosting process.

1.2.2 Multi-Modality Learning

Physiological evidence and analysis of biological systems also show that fusion of audio-visual information is useful to enhance human perception [3, 45, 69]. As will be shown in detail in Section 2.5, several works combine the audio information with the visual cues for speech/speaker recognition and object localization in videos. For example, visual features obtained by tracking the movement of lips, mouths, and faces can be combined with audio features describing acoustic speech for improved speech and speaker recognition [92, 141].

However, most of the previous multi-modality approaches cannot be easily applied to detect generic semantic concepts in general videos, due to several reasons. First, both object detection and tracking are difficult in general videos. There exist uneven lighting, clutter, occlusions, and complicated motions of multiple objects and the camera. As a result, it is hard to segment objects with satisfactory accuracy. Second, blind sound source separation, especially in real-world outdoor scenes, remains challenging. Finally, the synchronization between sounds and visual objects that make sounds cannot be observed most of the time. Objects may make sounds without large movements, and often some objects making sounds do not appear in the video.

1.2.2.1 Global Multi-Modal Fusion

Due to the above challenging conditions, the current concept detection methods (as will be shown in Section 2.5.3 in detail) take the global fusion strategies to use both audio and visual information, which avoid object-level visual and audio analysis or synchronization. In early fusion, features from different modalities are concatenated to make a long feature vector to train classifiers [245]. In late fusion, individual classifiers are built for each modality, and their predictions are combined to make the final decision [90, 130]. Although the fusion frameworks are simple to use, there are several issues limiting the performance. Early fusion methods usually suffer from the "curse of dimensionality", as the concatenated multi-modal feature can have very high dimensionality (*e.g.*, thousands of dimensions). In addition, different modalities may contribute to the classification task unevenly, *i.e.*, some modalities may be more important than others in detecting a specific concept. Dimension reduction and feature selection are usually required in practice, which are still unsolved in most practical cases. As for late fusion, how to select the appropriate classifier combination strategy remains a basic machine learning problem [108, 189] and the best combination strategy depends much on the particular problem in hand.

In this thesis, we develop two multi-modality fusion approaches [24]. The first method is a late fusion approach, where the above mentioned BCRF-CF algorithm proposed for multi-concept learning is extended to an *Audio-Visual Boosted Conditional Random Field* (*A-V Boosted CRF*) method to incorporate both audio and vidual-based detectors for enhanced classification. In the first layer of CBCF, individual concept detectors are built over visual and audio features, respectively, which are then applied to get visual-based and audio-based probabilistic estimations of concept labels, respectively. All these visual-based and audio-based initial estimation results are used as inputs in the second layer, and the detection results of each individual concept can be updated through the context-based model. Late fusion through our A-V Boosted CRF provides a natural way to combine individual judgements from the visual channel and audio channel. Instead of choosing combination strategies explicitly, predictions from different modalities are fused together through pursuing discriminative class separation.

The second method is an early fusion approach, where we extend the above mentioned kernel and classifier sharing algorithm in Section 1.2.1.2 to an *Audio-Visual Joint Boosting* (*A-V Joint Boosting*) method. In this extended approach, both the visual-based VSPM kernels and the audio-based kernels from acoustic analysis are put together into a big kernel pool, on top of which the SVM-based joint boosting algorithm can be used to select the optimal subset of kernels to share among various concepts. Instead of using direct feature concatenation, our A-V Joint Boosting method selects and fuse the optimal types of kernels generated from multiple modalities for enhanced classification.

Both A-V Boosted CRF and A-V Joint Boosting can be considered as natural combinations of multi-modality learning and multi-concept learning, and are flexible to incorporate other modalities (*e.g.*, textual). For instance, textual features can be used either to train individual concept detectors as additional inputs for the CRF or to build additional kernels to be shared by joint boosting.

1.2.2.2 Local Audio-Visual Atoms

Despite promising performance improvements, the fusion methods (either early or late fusion) have limitations in capturing joint audio-visual local patterns. The joint representation of local visual and audio features captures the correlations between audio and video features that are unique for videos of a certain concept. For example, the joint pattern of a cake object and the birthday music is an intuitive joint audio-visual cue for the "birthday" concept, while the joint pattern of a cake object and the wedding music strongly implies the "wedding" concept. Additionally, the joint pattern of smoke and the siren sound may suggest an explosion/fire incident, while the busy highway and the siren sound may indicate some traffic accidents. However, such audio-visual patterns have never been explored in general video analysis before.

As discussed before, object-level audio and visual analysis and audio-visual synchronization are difficult in general videos. In this thesis, we propose to capture interesting audio-visual cues by extracting localized *Audio-Visual Atoms* (*AVAs*) in videos [99]. AVAs provide a balanced choice for exploring audio-visual correlation in general videos. Compared to the tight audio-visual synchronization methods focusing on object detection and tracking in both audio and video, AVAs do not rely on precise object extraction. Instead, we track visually consistent regions in video and locate audio onsets in the audio soundtrack, respectively. Then we generate an audio-visual atomic representation in which a moderate level of synchronization is enforced between local region tracks and local audio onsets. The audio-visual correlation we capture is based on co-occurrence in a short-time window, *e.g.*, the co-occurrence of a cake and the birthday music or that of smoke and the siren sound. This is in contrast to the precise synchronization between tracked visual objects and audio sources in the sound track.

Specifically, we track automatically segmented regions based on the visual appearance within a short video slice (*e.g.*, 1 second), and then connect visually similar short-term region tracks from adjacent short-term video slices into long-term region tracks, called *visual atoms*. At the same time we locate audio energy onsets from the corresponding audio soundtrack by decomposing the audio signal into most prominent bases from a time-frequency representation, and then extract audio features from the reconstructed audio signal within a short window around each local energy onset. Such reconstructed short-term audio signals around energy onsets are called *audio atoms*. Then visual atoms and audio atoms are combined together to form joint audio-visual atomic representations, *i.e.*, AVAs. Based on these AVAs, joint *audio-visual codebooks* are constructed, and the codebook-based features are used for concept detection.

1.3 Unique Contributions

The contributions of this thesis can be summarized as follows. (1) An in-depth study of jointly detecting multiple concepts in general domains, where concept relationships are hard to determine. (2) The first system to explore the “20 questions” problem for semantic concept detection, by incorporating users’ interactions and taking into account joint detection of multiple concepts. (3) An in-depth investigation of combining audio and visual information to enhance detecting generic concepts. (4) The first system to explore the localized joint audio-visual atomic representation for concept detection, under challenging conditions in general domains.

We evaluate the proposed approaches over two large-scale realistic data sets in the general domain: the TRECVID 2005 news video set [170], and Kodak’s consumer benchmark video set [137]. Both data sets are among the largest and the most challenging ones existing for evaluating video concept detection methods. The experimental results can be summarized as follows.

The multi-concept approaches are evaluated over the TRECVID 2005 news video set. Experimental results confirm the effectiveness of the proposed BCRF-CF and the method predicting when it will work. That is, out of 39 evaluated concepts 26 are automatically chosen by our prediction method to use BCRF-CF, among which 21 actually get performance improvements; the overall MAP on these 26 concepts is improved by 6.8% compared to the baseline independent detectors. In addition, the Active CBCF paradigm can indeed improve concept detection by using user's annotation. For example, the MAP of Active CBCF on the selected 26 concepts outperforms BCRF-CF by 3.4%, and outperforms the baseline individual detectors by 9.2%. As for the proposed multi-concept approach with feature and classifier sharing, experimental results also demonstrate significant performance gains, *i.e.*, 10% in MAP and up to 34% AP for some concepts like "maps", "building", and "boat-ship".

The multi-modality approaches are evaluated over Kodak's consumer benchmark video set. Results show that compared with using individual visual or audio features alone, by combining audio and visual aspects using multi-modality fusion (A-V Boosted CRF or A-V Joint Boosting) we can get significant performance improvements, *e.g.*, a 10% MAP gain. For the localized joint audio-visual atomic representation, experiments confirm that it can capture frequently co-occurring audio-visual patterns unique to individual concepts, and achieve a significant performance improvement. For example, compared with single-modality approaches, the overall MAP is improved by 8.5%, and more than 20% AP gains are obtained for many concepts like "animal", "beach", "boat", "dancing", "museum", and "playground".

1.4 Thesis Overview

The remainder of the thesis is organized as follows. In Chapter 2, we provide a survey of research works related to semantic concept detection in images and videos and state-of-the-art solutions using multi-modality and/or multi-concept learning. In later parts of Chapter 2, we describe the experimental setup used throughout the whole thesis. Specifically we present Kodak's consumer benchmark video set. As far as we know, this is the first systematic work in the consumer video domain aimed at the definition of a large lexicon, construction of a large benchmark data set, and annotation of videos in a rigorous fashion.

After that, we introduce the TRECVID 2005 news video set, the performance evaluation metrics, the visual features we will use for most of our methods, as well as the baseline approach we will use for comparison.

In Chapter 3 we introduce the proposed BCRF-CF algorithm for multi-concept learning, under the framework of CBCF. In addition, we study the "20 questions problem" for semantic concept detection by developing the Active CBCF approach.

In Chapter 4 we describe the details of sharing kernels and classifiers through the construction of a multi-label concept detector by joint boosting. Our method automatically selects optimal kernels and subsets of sharing concepts in an iterative boosting process, which is also flexible in incorporating any type of kernel.

In Chapter 5 we introduce our multi-modality fusion algorithms: A-V Boosted CRF and A-V Joint Boosting. They incorporate both audio and visual signals for enhanced classification by extending, respectively, the BCRF-CF algorithm proposed in Chapter 3 and the kernel/classifier sharing method proposed in Chapter 4.

In Chapter 6 we describe our approach to discover joint audio-visual patterns in general videos. We extract atomic representations, *i.e.*, AVAs, based on which joint audio-visual codebooks are constructed to capture salient audio-visual patterns for effective concept detection.

Finally, in Chapter 7 we present the conclusions and future research work.

Chapter 2

Review of Concept Detection in Images and Videos

In this chapter we will provide an overview of research related to semantic concept detection for both static images and dynamic video sequences. These include image classification based on the global images; recognition of regions or objects in images; object or region-level action detection in video sequences; and concept detection over entire video sequences. In addition, we will review previous works exploring multi-modality learning that use both visual and audio information, and works that use context information for improving content analysis accuracy. We will also describe the data sets used in experiments, including Kodak's consumer video benchmark data set developed by us [137], the TRECVID news video set [170], and the performance metrics associated with these data sets.

2.1 Terminology

As shown in Fig. 2.1, a video concept detection task may be formulated as follows. Let $\mathbf{v}$ denote a video that is partitioned into K consecutive video segments $u_1, \ldots, u_K$ with fixed-length intervals or shot segmentation boundaries (shots are smooth video segments between transitions). A set of frames $\tilde{I}_k^1, \ldots, \tilde{I}_k^T$ are uniformly sampled (*e.g.*, 30 frames per second) from each video segment u_k. In addition, representative keyframes are extracted from each video segment u_k, *e.g.*, one keyframe I for each segment u_k. Assume that there are M semantic concepts, $C^1, \ldots, C^M$. Let $\mathbf{y}_\mathbf{x}$ denote the concept labels of an data $\mathbf{x}$, $\mathbf{y}_\mathbf{x} = [y^1_\mathbf{x}, \ldots, y^M_\mathbf{x}]$, with $y^m_\mathbf{x} = 1$ or -1 indicating the presence or absence of concept C^m in data $\mathbf{x}$. Data $\mathbf{x}$ can be a frame $\tilde{I}$, a keyframe I, a video segment u, or an entire video $\mathbf{v}$. Let $\mathcal{F}^v$ ($\mathcal{F}^a$) denote a certain feature space describing the visual (audio) characteristics of videos, and $\mathbf{f}^v(\mathbf{x})$ ($\mathbf{f}^a(\mathbf{x})$) is the feature vector of data $\mathbf{x}$ in the feature space $\mathcal{F}^v$ ($\mathcal{F}^a$).

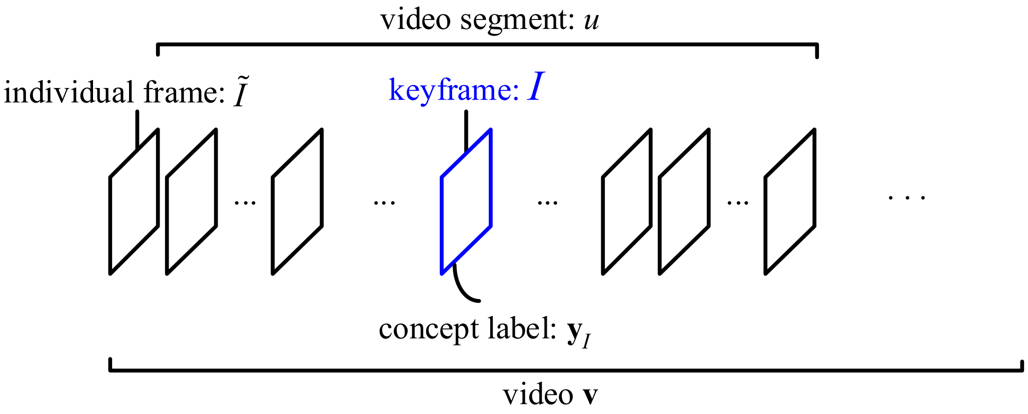


Figure 2.1: Data structure of a video concept detection task.

Video concept detection can be achieved in many different ways. We can classify individual image keyframes or classify the entire video sequence. Static image-based classification is a popular way for video concept detection [2, 24, 25, 170, 248, 249]. In this approach, classifiers are applied to static image keyframes sampled from video segments, which estimate likelihoods of the presence of concepts in each keyframe. Over the entire video segment, these concept occurrence likelihoods over individual keyframes are aggregated to generate the overall concept detection results. Since only keyframes are processed instead of the entire sequence of frames, the static image-based approaches are relatively fast. However, the temporal information is ignored, which limits the capabilities of these methods in modeling temporal-related concepts, such as events or object actions. To address such issues, several approaches have been proposed to incorporate temporal information [36, 44, 107, 117, 151, 205, 263]. However, due to the large scene variations in illumination, background, clutter and occlusion, these methods are only used in staged videos or surveillance applications with fixed (or almost fixed) cameras and stable backgrounds.

In the following sections we will review state-of-the-art concept detection techniques of classifying static images and dynamic video sequences. Specifically, we group the methods to two categories: those classifying entire images or video sequences globally, and others recognizing local object/regions or their actions.

2.2 Related Works on Image Classification

To classify whether a concept is present in an image, the following two elements are required. A set of visual descriptors (or features) are needed to describe the visual characteristics of images. Then a machine learning classifier is needed to predict concept presence using these visual descriptors.

2.2.1 Visual Descriptors

There have been many different types of visual descriptors developed to capture various visual characteristics. In this subsection we briefly summarize some of the most commonly used features for detecting generic visual concepts. These descriptors can be categorized as global ones over entire images or local ones over local patches.

2.2.1.1 Global Descriptors

Global descriptors such as color, texture, or edge histograms [167, 173, 190, 204, 217, 227], coherence vectors [183, 184, 227], correlograms [88, 136, 143], textures from band-pass filters [148, 149], and color moments [248, 249], have been frequently used in classifying large-scale image and video collections [2, 24, 248, 249]. To better use the spatial information in image, several features have been developed recently, such as the layout histogram and multi-resolution histogram [70], the *Markov Stationary Features* (*MSF*) [127], and the contextualized histogram [166]. For example, MSF adopts Markov chain models to characterize spatial relationship between histogram bins. It treats the bins as states in Markov chain models, and interprets the bin co-occurrence as the transition probability between states.

The main drawback of global visual features is their inability to model the individual objects in images. As a result, global visual features may not give satisfactory performance in classifying object-oriented concepts, such as “bike”, “dog”, “car”, *etc.* The major advantage of global descriptors, however, is their simplicity and efficiency since region segmentation or object extraction is not needed. Thus, global descriptors are still widely used for image classification, especially in scenarios involving a large number of images.

2.2.1.2 Local Descriptors

In recent years, local descriptors have become very popular for image classification and object recognition. Typically, a local descriptor is extracted from a local region or patch centered on a local interest point. The local interest point is an image pattern that differs from its neighborhood and is usually associated with a change of certain image properties such as intensity, color, and texture.

Tuytelaars and Mikolajczyk [226] categorize local interest points into three broad categories based on their possible usage. The first type has specific semantic interpretations in some restricted application context. For instance, edges detected in aerial images often correspond to roads; blob detection can be used to identify impurities in some inspection tasks; *etc.* The second type is the well localized and individually identifiable anchor points, whose location can be determined accurately and in a stable manner over time. For instance, the features used in the KLT tracker [207] minimize intensity dissimilarity for tracking under the assumption of affine transformation. Finally the third type is the interest points introduced for robust image representation and recognition without the need for segmentation. They do not have to be localized precisely, since the goal is not to match them on an individual basis, but rather to analyze their statistics in the local neighborhood of these points. The last type has become popular in recent image classification and object recognition research, including interest points based on image intensity, *e.g.*, Hessian points [11, 153, 154], Harris points [9, 73, 133, 154], *Maximally Stable Extreme Regions* (*MSER*) [152, 156], and *Laplacian-of-Gaussian* (*LoG*) or *Difference-of-Gaussian* (*DoG*) points [138, 200], and those based on image color, *e.g.*, salient points based on color distinctiveness [229] and Harris color points [157]. Different interest points emphasize different aspects of invariance of image properties. For example, LoG/DoG and Hessian detects target at blobs with local maxima/minima in the intensity or color channels. The Harris Laplace detector finds corner points, specifically, points with large variation of autocorrelation at different scales. Detailed description and discussion about these interest points can be found at [132, 226, 228].

Given the detected interest points, a large number of local descriptors can be calculated over the local regions that emphasize different image properties like pixel intensities, color, texture, edges, *etc.* The *Scale Invariant Feature Transform* (*SIFT*) descriptor proposed by Lowe [138] has been very popular for image matching and object recognition. The SIFT descriptor is represented by a 3D histogram of gradients (typically with 128 dimensions) computed from the local neighborhood based on locations and orientations. The invariance property of SIFT is usually inherited from the invariant interest points. For instance, SIFT is scale-invariant over DoG minima/maxima that are applied in scale-space to a series of smoothed and resampled images[1], and is affine-invariant over Harris-affine or Hessian-affine points that define affine neighborhoods by the affine adaptation process based on the second moment matrix. The quantization of gradient locations and orientations makes SIFT robust to small geometric distortions and small errors in region detection. Similar to the SIFT descriptor, the edge-based *shape context* proposed in [12] is a 3D histogram of edge point locations and orientations, which has been successfully used, for example, for shape recognition of drawings where edge features are reliable. However, this edge-based descriptor is very sensitive to image blur. Lazebnik *et al.* [118] propose an invariant descriptor called *spin image*, which is a histogram of pixels based on quantized locations and intensity values. They show that the region-based spin image is very effective in texture classification. However this descriptor is sensitive to illumination changes. The *Gradient Location and Orientation Histogram* (*GLOH*) descriptor proposed in [155] extends the location grid of SIFT and uses PCA to reduce the feature size, where the covariance matrix for PCA is estimated on local image regions collected from some training images. A comprehensive empirical comparison of several local descriptors is conducted by Mikolajczyk and Schmid and can be found in [155].

With the development of the large variety of local descriptors that have shown promising

[1] Dominant orientation is assigned to each local interest point to make SIFT rotation-invariance in [138].

performance, the *Bag-of-Features* (*BoF*) representation [54, 100, 235, 259] has attracted much attention in recent years. The BoF method treats an image as a collection of unordered local descriptors extracted from local regions, quantizes them into instances of discrete *visual words* defined in a *visual vocabulary* that is often constructed by an unsupervised clustering method like K-means. Many works have been done to improve the traditional BoF model, such as generative methods in [16, 18, 125, 193] for modeling the co-occurrence of the codewords, discriminative codebook learning in [50, 103, 158, 253] instead of standard unsupervised clustering, and *Spatial Pyramid Matching* (*SPM*) [119] for modeling the spatial layout of the local features. Among these extensions, SPM has been shown quite successful in the evaluation of several benchmarks like Caltech-101 [124] and Caltech-256 [67]. Also, a linear SPM is recently proposed by Yang *et al.* [252] based on SIFT sparse codes to further reduce the computational complexity.

2.2.2 Kernel-based Classification

Among various approaches for image classification, kernel-based discriminative learning methods, *e.g.*, the *Support Vector Machine* (*SVM*) [230], have been demonstrated effective for detecting generic concepts in many works [2, 17, 64, 65, 67, 85, 110, 119, 171, 259]. In this thesis, all of our proposed algorithms will also use kernel-based discriminative learning methods, specifically, SVM-based algorithms, for video concept detection.

2.2.2.1 SVM Classifier

The target of SVM [230] is to learn a classifier based on a set of labeled training data $\mathcal{X} = \{\mathbf{x}_n, y_{\mathbf{x}_n}\}$, $n = 1, \ldots, N$. Each training data $\mathbf{x}_n$ is represented as a point in a d-dimension feature space and is associated with label $y_{\mathbf{x}_n}$, where $y_{\mathbf{x}_n} = \pm 1$. SVM finds the classifier as an optimal separating hyperplane that gives a low generalization error while separating the positive and negative training samples. Given a data point $\mathbf{x}_n$, SVM determines its corresponding label by the sign of a linear decision function: $f(\mathbf{x}_n) = \mathbf{w}^T\mathbf{x}_n + b$. For learning non-linear classification boundaries, a mapping function ϕ is introduced to embed data vector $\mathbf{x}_n$ into a high dimensional feature space as $\phi(\mathbf{x}_n)$, and the corresponding class label is given by the sign of $f(\mathbf{x}_n) = \mathbf{w}^T\phi(\mathbf{x}_n) + b$. In the SVM classifier, this optimal hyperplane is determined by giving the largest margin of separation between different classes, *i.e.*, by solving the following problem:

$$\begin{aligned}&\min_{\mathbf{w},b,\epsilon} Q = \min_{\mathbf{w},b,\epsilon} \left\{ \frac{1}{2}||\mathbf{w}||_2^2 + C\sum_{n=1}^{N} \epsilon_i \right\},\\ &s.t.\ y_{\mathbf{x}_n}(\mathbf{w}^T\phi(\mathbf{x}_n)+b) \geq 1-\epsilon_n, \epsilon_n \geq 0, \forall\ (\mathbf{x}_n, y_{\mathbf{x}_n}) \in \mathcal{X}\end{aligned} \tag{2.1}$$

where $\epsilon = \epsilon_1, \ldots, \epsilon_N$ are the slack variables assigned to training samples, and C controls the scale of the empirical error loss the classifier can tolerate.

By introducing non-negative Lagrange multipliers $\alpha = \alpha_1, \ldots, \alpha_N$ and $\mu = \mu_1, \ldots, \mu_N$, we have:

$$\begin{aligned}&\frac{\partial Q}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum\nolimits_{n=1}^{N} \alpha_n y_{\mathbf{x}_n} \phi(\mathbf{x}_n)\\ &\frac{\partial Q}{\partial b} = 0 \Rightarrow \sum\nolimits_{n=1}^{N} \alpha_n y_{\mathbf{x}_n} = 0,\\ &\frac{\partial Q}{\partial \epsilon_n} = 0 \Rightarrow C - \alpha_n - \mu_n = 0, \quad n = 1, \ldots, N\end{aligned}$$

And then we can get the dual problem of Eqn. (2.1) as follows:

$$\max_{\alpha} Q_{dual} = \max_{\alpha} \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n \alpha_m y_{\mathbf{x}_n} y_{\mathbf{x}_m} \phi^T(\mathbf{x}_n)\phi(\mathbf{x}_m) \tag{2.2}$$

Through defining $k(\mathbf{x}_n, \mathbf{x}_m) = \phi^T(\mathbf{x}_n)\phi(\mathbf{x}_m)$ as a kernel, we can solve the *Quadratic Programming* (*QP*) problem in Eqn. (2.2).

There are several advantages of using SVM for image classification. SVM provides non-linear classification and avoids explicit definition of the function class for non-linear mapping. SVM has good generalization ability and can get reasonable performance with limited training data. In addition, SVM is flexible to incorporate a large variety of kernels from which we can select appropriate ones for solving different problems. More theoretic discussion and analysis about SVM can be found in [230].

2.2.2.2 Kernel Construction

The kernel trick [1] is adopted in SVM to enable the use of a linear classifier to solve a non-linear problem by mapping the original non-linear observations into a high-dimension space. With Mercer's theorem, a continuous, symmetric, positive semi-definite kernel function

$k(\mathbf{x}_n, \mathbf{x}_m)$ can be expressed as an inner product in a high-dimension space: $k(\mathbf{x}_n, \mathbf{x}_m) = \phi(\mathbf{x}_n) \cdot \phi(\mathbf{x}_m)$. In practice, a large variety of kernels that satisfy Mercer's conditions have been applied to different types of features for image classification[2]. These include traditional kernels such as *Radial Basis Function* (*RBF*), pyramid matching kernel [64], spatial pyramid matching kernel [119], proximity distribution kernel [134], chi-square kernel [17], *etc.*

The proliferation of the large variety of kernels has also motivated researchers to explore combinations of kernels. In [64, 119] multiple-levels of histogram intersection kernels are linearly combined, resulting in some of the best image classification performances. Although the fixed weighting makes sense intuitively, a more rigorous way to find optimal weighting is necessary. De la Torre *et al.* [38] learn the positive combination weights of normalized kernels using gradient descent search. Bosch *et al.* [17] learn weighting parameters using a validation set, over which an exhaustive search for the optimal weighting is pursued. Both methods cannot scale to a large number of kernels due to the complexity of the computation overhead.

To further improve the classification performance, recently researchers have started to explore the more general *Multiple Kernel Learning* (*MKL*) problem: how to effectively combine heterogeneous kernels constructed from different feature spaces; how to select the appropriate kernels for different tasks; how to determine the optimal weights to combine kernels; and finally, if possible, how to simultaneously learn an optimal kernel combination and the associated classifier. Some methods [5, 34, 110, 116] formulate the objective of learning a sparse combination of kernels as a convex, quadratically constrained quadratic problem, which can be efficiently solved, for example, by using semidefinite programming or sequential minimal optimization techniques. To incorporate information from unlabeled data, semi-supervised kernel matrix learning methods have also been developed [81, 264].

2.2.3 Evaluation

There are several data sets widely used to evaluate generic image classification, including Caltech-101 [124], Caltech-256 [67], the PASCAL *Visual Object Detection Challenge* (*VOC*)

[2] Some kernels that do not satisfy Mercer's condition have also been used in some applications, such as the image matching kernel in [19] for object recognition.

[182], Corel data sets [22] (*e.g.*, Corel5K [47] and Corel30K [126]), *etc.* The Caltech-101 data set [124] consists of images from 101 object categories and an additional background class. The images were collected by choosing a set of object categories, downloading data from Google Image and then manually screening out images that did not fit the category. The data set contains some variations in color, pose and lighting. Caltech-256 [67] was collected in a similar manner to Caltech-101, where the number of categories was increased to 256, the minimum number of images in each category was increased, the artifacts due to image rotation were avoided, and a new and larger clutter category was introduced for testing background rejection. Images in both Caltech-101 and Caltech-256 are often uniform in presentation, well aligned, and usually not occluded. As a result, the images are often quite different from real-world cases where there often exist clutter, occlusion, and variance in position, size, and orientation of objects and backgrounds [187].

The PASCAL VOC data consists of a few thousand images annotated with bounding boxes for objects of twenty categories (*e.g.*, “car”, “bus”, “airplane”). The classification challenge is the following: for each of the twenty classes, predict the presence/absence of any instance of the class in the test image. The participants will generate the confidence score of the predicted class for each test image during evaluation. Compared to Catelch-101 or Caltech-256, PASCAL VOC has a smaller number of categories. However, the number of images in each categories is larger, and there is much more intra-class variation. So far, the state-of-the-art image classification performances are achieved by using multiple types of local descriptors in the form of BoF, including information about spatial layout of local descriptors, and learning good combinations of multiple kernels [67, 119, 124, 182, 258].

2.3 Related Works on Object/Region Recognition in Images

Detection and localization of specific objects or regions can be very helpful for image classification. For instance, if one has a good object detector, it becomes easy to predict image categories when some objects are successfully detected. In this subsection we summarize state-of-the-art object and region-based recognition methods from two different perspectives: object localization and multi-class image segmentation, both of which have attracted great attention during the recent years in the computer vision field.

2.3.1 Object Localization

The objective of object localization is to determine if one or more objects of a category are present in an image and, if so, find the locations of such objects in the image. This is a challenging issue due to appearance variations, scene clutter, viewpoint changes, and deformations of the objects.

In object localization, various approaches have been proposed to locate objects in the scene by matching object models to the observations, including pictorial structures [53], constellation models [54], implicit shape models [121]. Search for the optimal matching is usually done either of the following methods: sliding windows [31, 35, 55, 234] and voting methods based on the Hough transform [86].

Sliding windows scan over possible locations and scales, evaluate a binary classifier, and use post-processing such as non-max suppression to detect objects. Sliding window classifiers are well suited for detecting rigid objects and have been widely used for detection of faces [178, 234], pedestrians [35, 159], cars [180], *etc.* The computational burden of this procedure is in general large although various techniques have been proposed to alleviate the complexity issue, such as cascaded classification [232, 234], branch-and-bound [115], and efficient classifier evaluation [144]. For example, cascades have several computational limitations: the training process is slow, and a cascade usually reduces the overall performance. In addition, as discussed in [68] the sliding window approaches differ significantly from the nature of human visual detection, where attention is directed to certain locations based on low-level salience rather than uniformly to all locations. Nonetheless, due to its simplicity, the siding window approach remains to be the most popular one for locating objects in scenes.

Hough voting [86] parametrizes the object hypothesis (*e.g.* the location of the object center), lets each part of the object model vote for a point in the hypothesis space, and finally chooses the hypothesis with the highest number of votes. The Hough transform has been used for various problems including shape detection [6], instance registration [138], and category recognition [145, 177, 208]. Hough transform is much faster than sliding windows (*e.g.*, analogous to linear sliding windows in [120]), while sliding windows methods are more flexible. That is, Hough transform only works on constrained models, *e.g.*, detecting certain shapes like lines, while sliding windows can work over discriminatively learned models. One key issue for Hough voting is how to find a local estimate of the scale, for which various approaches have been proposed in earlier years [104, 138]. However, recent works obtain good performance by sampling features densely over a large range of scales [16, 51]. When the Hough transform is extended to estimate both location and scale, each local feature casts votes that form lines through the hypothesis space instead of a single point as in most current voting methods [56, 122, 177]. Ideally, all points on an object would yield lines that intersect in a single point. Due to intra-class variation and background clutter, the points of intersection are degraded into scattered clouds. In [175] an effort has been made to search for these object clusters through a weighted pairwise clustering of local votes. However, finding these clusters is quite difficult due to the unknown cluster number (that corresponds to the number of objects in the scene) and the difficulty in assigning votes to objects.

2.3.2 Evaluation of Object Localization

Two most popularly used data sets to evaluate object detection algorithms are the PASCAL VOC data set [182] and the MSRC data set [210]. Again, the PASCAL VOC set consists of a few thousand images annotated with bounding boxes for objects of twenty categories. The detection challenge is the following: predict the bounding box and label of each object belonging to any of the target classes in a test image. Each bounding box has a confidence value, which is used to generate a precision-recall graph for each class during performance evaluation. Detections are considered true or false positives based on their overlap with ground truth bounding boxes. The overlap between a proposed bounding box R and the ground-truth box Q is computed as: $\frac{\text{area}|Q\cap R|}{\text{area}|Q\cup R|}$. An overlap of 50% or greater is labeled as true positive. Any additional overlapping bounding box (duplicate detections) are rejected as false positives. Performance is then measured by the average precision. More details of the challenge can be found in [182]. The MSRC data set [210] consists of 591 photographs of 21 object classes: "building", "grass", "tree", *etc.* The training images are hand-labeled with the distinct object classes. The data set consists of general lighting conditions, camera viewpoint, scene geometry, object pose and articulation. More details can be found in [210]. So far, similar to the case of image classification, the state-of-the-art performance of object localization is achieved by using BoF with multiple types of local descriptors, including spatial layout information of local descriptors. More detailed discussion of the state-of-the-art performance over the PASCAL VOC challenge can be found at [52].

2.3.3 Multi-Class Image Segmentation

Segmenting an entire image into distinct recognizable regions is a central challenge in computer vision that has received increasing attention in recent years. Unlike object localization methods that aim to find a particular object, multi-class image segmentation (also called multi-class image labeling) methods aim at concurrent multi-class object recognition and attempt to classify all pixels in an image.

A number of methods have been proposed to address this problem [63, 75, 109, 194, 210, 239, 240]. Typically these algorithms construct *Conditional Random Field* (*CRF*) [114] defined over the pixels (or small coherent regions called superpixels) with local class predictors based on pixel appearance and a pairwise smoothness term to prefer consistent labels over neighboring pixels (*i.e.*, classifying visually-contiguous regions consistently). Optimizing the energy defined by this CRF is then equivalent to finding the most probable segmentation. The superpixel representations are usually obtained from unsupervised segmentation algorithms, and such representations have the benefit that features computed from constituent pixels in the segment can be computed and used for classification. Furthermore, it is also faster as inference only needs to be performed over a small number of superpixels rather than all the pixels in the image.

Due to imperfect segmentation, image segments are usually inconsistent with object boundaries. To alleviate these problems [82] and [199] use multiple segmentations of the image (instead of only one) hoping that although most segmentations are inaccurate, some are correct and thus useful for recognition. The recognition results of the multiple segmentations are merged using heuristic algorithms. In [109], higher order CRFs are used whose potentials are defined on sets of pixels (image segments) as a generalization of the commonly used pairwise smoothness potentials. The graph cut based algorithms are then used to conduct inference under such a framework.

Besides object detection and multi-class image labeling, there are some interesting works targeting at high-level holistic scene understanding. The aim is to jointly reason about objects, regions and 2D geometry of a scene, hoping that the errors from separate tasks can be reduced through the holistic modeling. An important step towards the goal of holistic scene understanding is to decompose the scene into regions that are semantically labeled and placed relative to each other within a coherent scene geometry. The global scene information has been incorporated in several works for improving object recognition or detection [83, 161, 218, 225]. New developments include the recent work of Tu [224] where a scene is decomposed into regions, text and faces using an innovative data driven MCMC approach on a generative model of the scene; the work of Li and Li [128] where a graphical model of events (*e.g.*, badminton or sailing) in images is proposed to describe each event as a latent factor conditioning the generation of objects and scene categories; and the latest work of Li *et al.* [129] where a hierarchical generative model is used to classify the overall scene, recognize and segment object components, and annotate the image with a list of tags.

Researchers have used many different data sets to evaluate multi-class image segmentation algorithms, including the MSRC data set [210], the 7-class Corel and Sowerby data sets [75], the 3-class GC data set [84], *etc.* However, there still lacks systematic evaluation to compare different multi-class image segmentation algorithms over a unified benchmark. This is partly because of the difficulty in acquiring a large amount of ground-truth data for evaluation, where images need to be fully segmented and image regions need to be annotated, both manually.

2.4 Related Works on Classifying Dynamic Video Sequences

In order to detecting temporal-related events and object actions, many methods have been proposed to incorporate temporal information from dynamic video sequences. Similar to static image-based classification, these methods can be roughly categorized into those recognizing object-level actions (*e.g.*, "people walking", "car exiting" and "people carrying a luggage") or others classifying global video sequences into generic semantic categories (such as "parade" and "wedding").

2.4.1 Object-Level Action Recognition

Robust detection and recognition of object-level actions is very useful for video content description, since it provides unique object-level dynamic information. As summarized in [165], actions may involve dynamic textures (*e.g.*, "flowing water"), activities that are temporally periodic and spatially restricted (*e.g.*, "people walking"), or motion events that do not repeat (*e.g.*, "people smiling"). Object-level action recognition is a challenging task, not only because of the scene variability in illumination, background, occlusion, but also because of object motion, deformation, and interaction among objects. In the following we briefly review previous object action recognition approaches. These methods have been widely used in the surveillance, indoor and office environments, and sports domains, in which videos are often captured by fixed (or almost fixed) cameras.

Earlier works for object action detection are constrained to well-controlled environments. For example, Bobick and Davis [15, 37] derive the temporal template representation from background subtracted images. They can handle a variety of choreographed actions across different subjects and views, but require two stationary cameras and a stationary background. Blank *et al.* [14] represent actions by describing the spatio-temporal shape of silhouettes. Their approach relies on background subtraction to obtain a silhouette of the moving subject. Izumi and Kojiama [93] propose an approach to generate natural language descriptions of human behavior. A hierarchy of actions called a case frame is developed for each body part and each object. Zelnik-Manor and Irani [256] use marginal histograms of spatio-temporal gradients at several temporal scales to cluster and recognize video events. Rao *et al.* [197] use the motion curvature of objects and conduct dynamic object segmentation in space and time. The time, location and sign of the curvature are matched to templates for action recognition. Ramanan *et al.* [196, 206] track objects through time and generate a trace of model parameters, which is then compared with the trace of the target spatio-temporal pattern to determine whether the observed event is of interest. Efros *et al.* [49] study the actions in sports games with relatively simple backgrounds. Ke *et al.* [107] proposed the volumetric features to correlate spatio-temporal shapes to segmented video clips. Albeit intuitively reasonable, these methods have difficulty in applying to videos where camera motion cannot be robustly removed and clean foreground objects cannot be extracted.

Recently, spatio-temporal interest points have become popular in the action recognition community. Both generative and discriminative methods have been developed. For example, Dollár *et al.* [43] model neighborhoods of interest points as cuboids and construct histograms based on normalized pixel intensity, brightness gradient and windowed optic flow. Li *et al.* [169, 201] propose a generative action recognition model using (unsupervised) *Latent Dirichlet Allocation* (*LDA*) or *probabilistic Latent Semantic Analysis* (*pLSA*) latent topic models and features proposed in [43] as well as correlograms. Hu *et al.* [87] address human action recognition in complex scenes where they employ the multiple instance learning SVM algorithm to alleviate the ambiguity problem where both spatial location and time period of the action cannot be precisely located. Laptev *et al.* [117] use spatio-temporal grids extracted at multiple scales to compute the *Histogram of Oriented Gradient* (*HOG*) [36] and the *Histogram of Optic Flow* (*HOF*) within each volumetric cell, based on a spatio-temporal BoF representation. Many efforts have been made to augment spatio-temporal interest points with additional information, *e.g.*, hierarchical structures [95], implicit shapes [242], local contexts [244], 3D spin images [135], 3D cube [255], and scene descriptors from object detection [72]. For example, in [72] the bag-of-detector scene descriptors are introduced for human action recognition, which encode presence/absence and structural relations between object parts and complement existing low-level video features. In general, spatio-temporal interest points capture distinctive local patterns in the spatio-temporal space where the difficult task of spatio-temporal object segmentation can be avoided. Also, the sparsity of interest points and their resulting computational efficiency is very appealing. However, spatio-temporal interest points focus on the local attributes at the risk of ignoring global information (such as global motion), and the detection of such interest points in complex scenes often fail in cluttered backgrounds.

2.4.2 Evaluation of Action Recognition

There are several public data sets generally used for evaluating object action recognition algorithms. The earlier action data sets, such as KTH [205], Weizmann [14], and the UCSD mice behavior [43] data, were collected in well-controlled environments, with clean backgrounds. Each video clip in these data sets involves only one type of action (*e.g.*, “eating”, “running” or “jogging”) and only one object, repeating the same action within the whole video clip. The CMU action videos [107] are also widely used for action recognition, where the videos were captured with a hand-held camera in crowded environments with moving people or cars in the backgrounds. There are five different types of human actions in the database, including “jumping jacks”, “pick-up”, “two-handed wave”, “one-handed wave” and “pushing elevator buttons”. There are relatively large variations in the actions performed by the subjects, with cluttered backgrounds and occluded actions. However, there exists a considerable gap between these staged samples and real-world scenarios where there are cluttered backgrounds or occluded crowds and the actions usually happen only once within a short duration. In real-world scenarios, it is very difficult to locate objects precisely, and it is hard to decide the start or end point of object actions of interest, or even the duration of each action. Recently several realistic data sets have been developed. One frequently used data set is introduced by Laptev *et al.* [117]: the Hollywood-1 movie data set, which is significantly more difficult than the previous data sets. This film data set contains eight different action classes: “answering the phone”, “getting out of the car”, “hand shaking”, “hugging”, “kissing”, “sitting down”, “sitting up”, and “standing up”. These actions were collected from 32 Hollywood movies. Lately, the Hollywood-1 Movie data set is further enlarged to Hollywood-2 movie data set [151] where four new actions are added: “driving car”, “eating”, “fighting person” and “running”. Twelve action classes are extracted from 69 Hollywood movies, where the total length of action samples is about 600 thousand frames. Each action clip usually lasts for a few seconds. In general, the current state-of-the-art performance for action recognition is achieved by fusing BoF representations built from multiple types of 2D and 3D local descriptors.

2.4.3 Generic Concept Detection by Classifying Video Sequences

Most of the above mentioned research on object-level action recognition is limited to videos captured by fixed (or almost fixed) cameras in surveillance applications or highly constrained videos. A more challenging problem is to recognize generic concepts in general domains, such as broadcast news videos, consumer videos like the ones on the internet (*e.g.*, on the YouTube web sharing site). In such cases, there are often fast moving small objects, large camera motion, articulate motions, significant clutter and object occlusion, and the event of interest may involve high-level semantic (*e.g.*, presidential campaign or wedding ceremony). There are some existing works, though limited, trying to address this challenging issue. Ebadollahi *et al.* [48] propose to treat each frame in a video clip as an observation and apply HMM to model the temporal evolution patterns of mid-level concepts in news videos. Dong and Chang [44] use kernel-based discriminative classification to detect a large variety of generic events in news videos, where the kernel is defined based on the BoF representation of temporal streams, with similarity metric computed using multi-level temporal matching. Zhou *et al.* [263] propose a SIFT-Bag based framework where each video clip is encoded as a bag of SIFT feature vectors, and the distribution of each video clip is modeled by *Gaussian Mixture Models* (*GMMs*). Then a SIFT-Bag kernel is calculated based on *Kullback-Leibler* (*KL*) distances of video GMMs that are used for discriminative classification. All these methods avoid object detection and tracking that are very difficult in these general videos, and optimize the detection accuracy directly by discriminative learning.

2.5 Related Works on Multi-Modality Learning

In the previous sections, we have briefly reviewed works related to classifying concepts in static images or dynamic video sequences. Most of these methods are developed based on visual descriptors that capture the visual characteristics of images and videos. Videos comprise of information from multiple modalities, both visual and audio. Each modality brings some information complementary with the other and their joint processing can help uncover relationships that are otherwise unavailable. Also, physiological evidence and studies of biological systems show that fusion of audio and visual information is key to human perception [3, 45, 69]. Thus, combining evidences from both visual and audio modalities has been considered increasingly important in recent years. In this section, we summarize methods that fuse audio and visual information in several tasks, such as speech and speaker recognition, object localization, and video classification.

2.5.1 Audio-Visual Speech and Speaker Recognition

Audio-visual speech and speaker recognition are among the earliest areas for joint audio-visual video analysis. In audio-visual speech recognition [46, 78, 92, 106, 191] or speaker recognition [60, 141, 237, 254], visual features obtained by tracking the movement of lips, mouths, and faces, are combined with audio features extracted from acoustic speech signals. Both *early fusion* (*i.e.*, feature fusion) and *late fusion* (*i.e.*, decision fusion) have been studied. Early fusion methods train a single classifier on the concatenated features, sometimes with additional transformation. Late fusion combines the output of classifiers separately developed using audio and visual modalities respectively.

2.5.2 Audio-Visual Synchronization for Localization

The task of multimodal localization can be described as follows: within the signal streams originating from one modality, localize the components that best correlate with the other modality in sapce, time, or both. For example, we would like to localize visual events corresponding to special sound events in the soundtrack, or identify the specific objects that produce the sound.

Many earlier works require the use of several microphones (*e.g.*, the "two-microphone and one camera" configuration), where stereo triangulation indicates the spatial location of the sounding sources [10, 28, 105, 203, 233, 265, 266]. A typical scenario is a known environment (mostly indoor), augmented with fixed cameras and microphones. Then moving sound sources (*e.g.*, persons) can be located by using both the audio signal time delays among the microphones and the spatial trajectories performed by the objects. These methods have achieved some success in several applications such as speaker localization in teleconferencing settings. However, in general scenarios videos are captured with a single camera where a video stream is associated with only a single sound track. In such cases, audio spatialization is no longer explicitly recoverable, and alternatively solutions are needed to correlate audio events with video objects. A common localization approach is to project each modality into a 1-D subspace [7, 79, 123, 212] and then correlate the 1-D representations. For example, Hershey *et al.* [79] localize the sound source in the video (*i.e.*, the speaker) by tagging the region of the image that has high temporal correlation with the audio information. The audio-visual synchrony is measured by the mutual information between the audio signal and individual pixels in the video modeled by a joint multivariate Gaussian process. Slaney and Covell [212] optimize temporal alignment between audio and visual tracks using canonical correlations. Under the assumption that fast moving pixels make big sounds, Barzelay and Schechner [7] locate sounding pixels by maximizing the correlation between the audio and visual temporal patterns that encode significant changes in the audio and visual signals, respectively. Later on, Fisher and Darrell [58] propose a more general approach, where they use the probabilistic multimodal generation model to measure mutual information between audio and visual signals based on non-parametric statistical density. Smaragdis and Casey [213] treat both the audio and visual streams as one set of data, from which independent components in some subspace are extracted. Ideally, these components correspond to objects in the scene that have simultaneous audio-visual presence. In [33] by visual and audio background/foreground (BG/FG) modeling, separate visual and audio patterns can be detected from a scene. Then the audio-visual integration is performed by studying audio-visual synchrony to detect audio-visual events such as “make a call” and “enter an empty lab”.

2.5.3 Multi-Modal Fusion for Video Classification

Most of the above mentioned multi-modality learning approaches cannot be easily applied to general videos due to several reasons. Visual object detection and tracking are known to be extremely difficulty in general videos. As a result, it is very hard to segment objects with satisfactory accuracy based on the current visual analysis techniques. Also, blind sound source separation, especially in real-world outdoor scenes, remains to be a challenging issue. The mixed sound track of general videos are usually generated by multiple sound sources under severe noises. Using the current audio analysis techniques, it is very difficult to extract individual sounds. In addition, the audio-visual synchronization cannot be observed most of the time. Multiple objects usually make sounds together in the video (*e.g.*, orchestra performance), with or without large movements, and often some objects making sounds are not visible in the video.

As a result, most existing methods for concept detection do not use direct correlation or synchronization across modalities. Instead, multi-modal fusion methods like those mentioned in Section 2.5.1 are used. For example, early fusion is used in [245] to concatenate features into long vectors. However, this approach usually suffers from the "curse of dimensionality", as the concatenated multi-modal feature can be very long. In addition, different modalities may contribute to the classification task unevenly, *i.e.*, some modalities may be more important than others. Dimension reduction and feature selection are required in practice. It remains an open issue how to construct suitable joint feature vectors comprising of features from different modalities with different time scales and different distance metrics.

In late fusion, individual classifiers are built for each modality separately, and their judgements are combined to make the final decision. Several combination strategies have been used to combine the individual judgements, *e.g.*, majority voting, linear combination, winner-take-all methods, HMM-based combination [90], super-kernel nonlinear fusion [245], or SVM-based meta-classification combination [130, 131]. However, classifier combination remains to be a basic machine learning problem [108, 189] and the best strategy depends much on the particular problem.

2.6 Related Works on Multi-Concept Learning

Traditional concept detection methods classify individual concepts independently. That is, an individual detection model (based on single modality or multiple modalities) is built for each concept to distinguish this concept from others. However, in the real world, semantic concepts do not occur in isolation. Such inter-concept context information can be used to enhance classification of individual concepts. For example, a confident detection of the "car" concept provides the evidence that increases the likelihood of detecting another concept "road" [61, 112]. Context information has been used by many previous works for image labeling, object localization, and for classifying generic concepts. In the following we review such methods in two broad categories: approaches for object and region-based recognition, and those for classifying images and videos.

2.6.1 Object and Region-based Recognition with Context Information

Context has been widely used for image labeling and object localization [21, 32, 74, 76, 83, 112, 161, 195, 211, 222]. There are different levels of contexts that can be used. For example, in the image labeling problem, context can model the local smoothness of pixel labels and the spatial relationships among image regions (*e.g.*, sky usually above water). For object localization, context can help model the co-occurrence relationship among objects (*e.g.*, a computer co-occurs with a keyboard) and the spatial relationship of objects (*e.g.*, a cup on top of a table).

2.6.1.1 Contextual Modeling for Image Labeling

An early work in [211] by Singhal *et al.* labels each region in the scene sequentially based on the labels of the previous regions. In recent years, approaches based on CRFs [114] have become popular [74, 111, 112, 160, 209, 210, 222, 224]. These methods typically treat the problem as finding a joint labeling for a set of pixels, superpixels, or image segments and are formulated as a CRF. Such CRFs for pixel or segment labeling use single-node potential features that capture local distributions of color, texture, or visual words. Compatibility potentials incorporate the labelings of neighboring pixels or segments. In particular, the two-layer hierarchical formulation proposed in [112] exploits different levels of context information, each layer being modeled as a CRF. Both the pixelwise smoothness context as well as the spatial relationships between image regions are modeled explicitly. Learning such complicated potentials is difficult and recent works mostly rely on boosting [209, 210, 222, 224] to select potential functions from a large space. For instance, Tu [224] proposes an auto-context algorithm addressing the posterior probability directly in a supervised manner, where the image appearance and context information are integrated by learning a series of classifiers.

2.6.1.2 Contextual Modeling for Object Localization

There are several types of context information that can be used to help object localization. The basic idea is that a confident detection of one object (*e.g.*, a car) provides the evidence that increases the likelihood of detecting another object (*e.g.*, road) [61, 112]. In addition, context information can be used within an object category, *e.g.*, a crowd of pedestrians reinforcing each other's detection. A recent paper [42] presents an empirical evaluation of the role of context in object detection.

Context can be used to model scene properties of the entire image [83, 220], the relationships between objects [21, 112], or the background surrounding the candidate objects [32]. For example, some works explore the co-occurrence of different objects in a scene [195, 221], explicitly learn the spatial relationships between objects [57, 61, 241], or learn the spatial relationships between an object and its neighboring regions [32, 76]. Using the co-occurrence context, Rabinovich *et al.* [195] show that the presence of a certain object class in an image is probabilistically related to the presence of another object class. Torralba *et al.* [221] assume that certain objects occur more frequently in certain scenes, *e.g.*, "tables" are more likely to be found in an "office" than on a "street". Many approaches explore the spatial relationships between objects. For example, Wolf *et al.* [241] detect objects using a descriptor with a large support range, allowing the detection of the object to be influenced by surrounding image pixels. Fink and Perona [57] use the output of boosted detectors for other concepts as additional features for the detection of a given concept. Murphy *et al.* [161] use the global "gist" feature to estimate the statistical priors on the locations of objects within the context of specific scenes. Another approach for modeling spatial relationships is to use the random fields such as the CRF to explicitly encode preferences for certain spatial relations [21, 112].

Despite the improvements by contextual modeling, the general problem remains to be a challenging one. Context information can be ambiguous and unreliable, and may not always be useful for object detection, especially when test data is from sources different from the training set where context information is obtained. In a recent work, Zheng *et al.* [261] attempt to use a discriminative maximum margin model to predict the effectiveness of context and to selectively apply context. However, how to evaluate the usefulness of different types of context information in a robust manner is still challenging.

2.6.2 Image and Video Classification with Context Information

In view of the difficulty in object segmentation in general videos, a holistic representation is usually preferred instead of the object/region-level approaches. Therefore, context models that capture the relationship among semantic concepts have been shown to be promising.

Several context-based algorithms have been proposed to classify generic concepts in large-scale general videos, such as TRECVID news videos [170] and Kodak's consumer videos [137]. An important branch of such methods is CBCF that usually has two layers. In the first layer, independent concept detectors are applied to extract posteriors of individual class labels on a given data, and then in the second layer detection results of individual concepts are updated through a context-based model that considers detection confidences of related concepts and the inter-conceptual relationships. Earlier CBCF approaches use pre-defined concept relations to improve the detection accuracy of individual classifiers such as the ontology hierarchy in [246] and the Bayesian networks in [179]. In [214], the vector model method does not explicitly model concept relationships. Instead, initial concept detection probabilities are used to form a concept space, within which discriminative classifiers are learned to refine detection results. The most popular choice is the graph-based methods where graphical models are used to encode the context information. Typically, graph nodes are concepts, the single-node potentials are independent concept detectors, and the two-node compatibility potentials reflect the pairwise conceptual relations, *e.g.*, the co-occurrence statistical relationships of concepts. The concept detection results can be refined through graph inference. For example, the Multinet method [162] uses a factor graph where the co-occurrence statistics of concepts are used as compatibility potentials. Posterior probabilities of concepts are updated by loopy probability propagation. Yan *et al.* [247] use several graphical models including directed and undirected ones (*e.g.*, Bayesian network, Markov Random Field, CRF, *etc.*), where co-occurrence statistics of concepts are used to compute compatibility potentials. Jiang *et al.* [102] propose a *Domain Adaptive Semantic Diffusion* (*DASD*) algorithm, to exploit semantic conceptual context while considering the change of context across domains. An affinity graph is constructed based on concept co-occurrences. Then semantic diffusion and graph adaptation are performed through alternatively updating concept affinity relations and detection results. An interesting perspective of DASD is that the contextual concept relations can be adaptively updated using the test data without relying on training data from the old domain. Similar to graph-based CBCF methods, there are some approaches that use graphical models to encode inter-conceptual relations, with the difference that the graph structure and concept detectors are learned together in a single step. For example, Qi *et al.* [192] propose a correlative multi-label method where concept relations are modeled by a Gibbs random field. However, due to the complexity of the learning algorithm (at least quadratic to the number of concepts), this method becomes infeasible for practical applications when the number of concepts is large (*e.g.*, dozens or hundreds).

CBCF, especially the graph-based approach, provides an intuitive and general framework for incorporating inter-conceptual relationships. It can be used to include diverse types of classifiers as single-node detectors and different types of prior knowledge in building graph structures. The main issue of graph-based CBCF methods, however, lies in the difficulty of designing appropriate compatibility potentials. Previous methods use co-occurrence statistics of concepts to approximate pairwise relations. Such approximations become unreliable when we only have limited training data. It is especially difficult to obtain accurate co-occurrence statistics of a diverse pool of generic concepts for general videos, due to the lack of prior knowledge about the concept ontology and insufficient annotation in practice. The error associated with the inaccurate approximation often leads to poor results in graph inference.

2.7 Data Sets and Performance Measures

In the thesis, we use two general video sets to evaluate different algorithms, namely, Kodak's consumer benchmark video set developed by us [137] and the TRECVID 2005 news video set [170]. In this section, we introduce these two data sets, followed by the performance evaluation metrics, and the baseline approach we use for comparison.

2.7.1 Construction of Kodak's Benchmark Consumer Video Set

To provide a sound foundation for developing and evaluating large-scale learning-based semantic indexing techniques, it is important to apply systematic procedures to establish large-scale concept lexicons and annotated benchmark video data sets. Recently, significant developments for such purposes have been made in several domains. For example, NIST TRECVID [170] has provided an extensive set of evaluation video data set in the broadcast news domain. Caltech-101 [124] and Caltech-256 [67] include images downloaded from the web to evaluate performances of image classification techniques. ImageCLEF [91] includes a large set of medical images and web images for evaluating image retrieval methods.

However, for consumer videos, to the best of our knowledge, there has been little effort so far to develop large-scale concept lexicons and benchmark data sets. Although automatic consumer video classification has been reported in the literature, most of the prior research deals with few concepts and limited data sets only. To contribute to the research of consumer video indexing and annotation, we have developed Kodak's consumer video benchmark data set, including a significant number of videos (a few thousand) from actual users who participated in an extensive user study over a one-year period. It also includes a lexicon with more than 100 semantic concepts and the annotations of a subset of concepts over the entire video data set. The concepts have been chosen in a systematic manner, considering various criteria discussed below. As far as we know, this is the first systematic work in the consumer domain aimed at the definition of a large lexicon, construction of a large benchmark video set, and annotation of videos in a rigorous fashion.

It is non-trivial to determine the appropriate lexicon of semantic concepts for consumer videos, as the correct lexicon may depend highly on the application. To fulfill the needs of actual users, we adopt a user-centric principle in designing the lexicon. The concepts are chosen based on findings from user studies confirming the usefulness of each concept. In addition, we consider the feasibility of automatic detection and concept observability in manual annotation. Our lexicon is broad and multi-categorical, including concepts related to activity, occasion, people, object, scene, sound, and camera operation. It also includes concepts manifested by multi-modal information. That is, our concepts may be visual-oriented and audio-oriented. To ensure the quality of annotation, we adopt a multi-tier annotation strategy for different classes of concepts. Some concepts use keyframe-based approaches to maximize the annotation throughput. For others, playback of an entire video clip is required to judge the presence of the concepts.

2.7.1.1 Lexicon and Concepts

The lexicon used in Kodak's consumer video benchmark data set was constructed based on an ontology derived from a user study conducted by Eastman Kodak Company. The ontology consists of 7 categories: subject activity, orientation, location, traditional subject matter, occasion, audio, and camera motion. Under these categories, over 100 concepts are defined based on feedback from user studies confirming the usefulness of each concept. The full list of categories and concepts can be found in [250]. This ontology has been chosen through three steps. First, an earlier user study based on a large collection of consumer photos has been conducted by Eastman Kodak Company to discover concepts interesting to users in practical applications. These concepts are used to form the initial candidate lexicon for the consumer video data set. Second, the initial concept list was refined based on a smaller-scale user study to find interesting concepts for consumer videos. A relatively smaller collection of video data, compared to the photo collection, was used. Finally, the availability of each selected concept (the number of videos we may obtain from users for each concept) is investigated, and the rare concepts are excluded.

Because of the limitation of both the annotation and the computation resources, 25 concepts are further selected from Kodak's ontology based on 3 main criteria: (1) visual or audio detectability – whether the concept is likely to be detected based on the visual or audio features; (2) usefulness – whether the concept is useful in practical consumer media application; (3) observability – whether the concept is observable by the third-person human annotators through viewing the audio-video data only. In addition, we also consider one additional criterion, availability, *i.e.*, the number of video clips we may expect to acquire for a concept from actual users. To estimate the availability of a concept, we conduct searches using concept names as keywords against YouTube and AltaVista, two popular sites for sharing user-generated videos. The number of returned videos in the search results is used to approximate the availability of a concept.

The final lexicon used in Kodak's consumer video benchmark data set contains 25 concepts as shown in Table 2.1. Note these concepts are multi-modal in nature – some are primarily manifested by the visual aspect (*e.g.*, "night", "sunset"), some are audio-oriented (*e.g.*, "music", "singing"), and others involve both visual and audio information (*e.g.*, "wedding" and "dancing").

2.7.1.2 Video Data Set and Keyframes

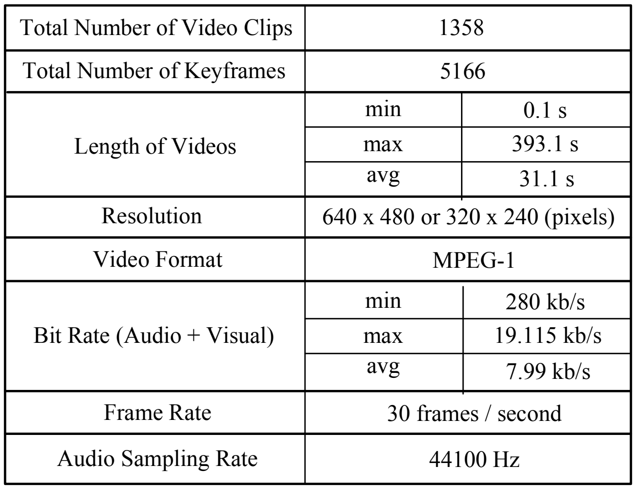
Kodak's video data set was donated by actual users to Eastman Kodak Company for research purposes. The majority of the videos were recorded by either the Kodak EasyShare C360 zoom digital camera or the Kodak EasyShare V570 dual lens digital camera. The videos were collected over the period of one year from about 100 users, thus spanning all seasons and a wide variety of occasions. It is also geographically diverse as the majority of users took videos outdoors and away from home, including trips across the US and also overseas. These users were volunteers who participated in typically three-week-long camera handouts. They represent different consumer groups (*e.g.*, proactive sharers, conservative sharers, prints-for-memory makers, digital enthusiasts, and just-the-basics users) and were identified through a web-based survey. Female users slightly outnumbered male users. A unified video format, MPEG-1, is used to allow easy handling of videos. The videos whose original format is QuickTime movie or AVI format were transcoded to MPEG-1 format according to original bit rates. Other detailed information is shown in Table 2.2.

From the videos, we sample keyframes based on a uniform time interval, *i.e.*, 1 keyframe per 10 seconds. Based on the experience obtained from the user study, we consider the 10-sec sampling interval to be a good tradeoff between computation requirements and indexing accuracy. For static concepts (*e.g.*, locations and occasions), we assume that the video content will not change much in each 10-sec interval. In such a case, keyframes are sufficient for analyzing the concept. However, for other concepts (*e.g.*, "dancing"), information in the temporal dimensions (object movements and dynamics) needs to be considered. In this case, features need to be extracted from image frames at a much higher rate. In other words, the above-mentioned coarsely sampled keyframes are more intended for analyzing the static concepts only. Concepts involving strong temporal information need to be analyzed using

Table 2.1: Selected concepts and definitions.

Concept		Definition
activities	dancing	One or more people dancing
	singing	One or more people singing. Singer(s) both visible and audible. Solo or accompanied, amateur or professional
occasions	wedding	Videos of the bride and groom, cake, decorated cars, reception, bridal party, or anything relating to the day of the wedding
	birthday	This event is typically portrayed with a birthday cake, balloons, wrapped presents, and birthday caps. Usually with the famous song
	graduation	Caps and gowns visible
	ski	Emphasize people in action (vs. standing)
	picnic	Video taken outdoors, with or without a picnic table, with or without a shelter, people, and food in view
	show	Concerts, recitals, plays, and other events
	parade	Processing of people or vehicles moving through a public place
	sports	Focus initially on the big three: soccer, baseball/softball, and football
	playground	Swings, slides, etc. in view
	park	Some greenery in view
	museum	Video is taken indoors and is of exhibitions of arts, crafts, antiques, etc.
scenes	sunset	The sun needs to be in front of the camera (although not necessarily in view)
	beach	Largely made up (1/3 of the frame or more) of a sandy beach and some body of water (e.g., ocean, lake, river, pond). Note “beach” should be explicitly called out. In a more strict definition, a “beach” scene contains at least 10% each of water, sand, and sky, and was taken from land. Pictures taken primarily of water from a boat should be called “open water”
	night	The video is taken outdoors at night (after sunset)
objects	people-1	One person: the primary subject includes only one person
	people-2	Group of two: the primary subject includes two people
	people-3	Group of three or more: the primary subject includes three or more people. This description applies to the primary subject and not to incidental people in the background
	animal	Pets (e.g., dogs, cats, horses, fish, birds, hamsters), wild animals, zoos, and animal shows. Animals are generally “live” animals. Those stuffed or mounted (taxidermy) may qualify depending on how “lively” they look
	boat	Boat in the water
people	crowd	The primary subject includes a large number of people in the distance
	baby	Infant, approximately 12 months or younger
sound	music	Clearly audible professional or quality amateur music in the soundtrack (which may also include vocals and other instruments). There is emphasis on the quality of the music
	cheer	One or more people cheering - shouts of approval, encouragement, or congratulation

Table 2.2: Information of Kodak's video data set.



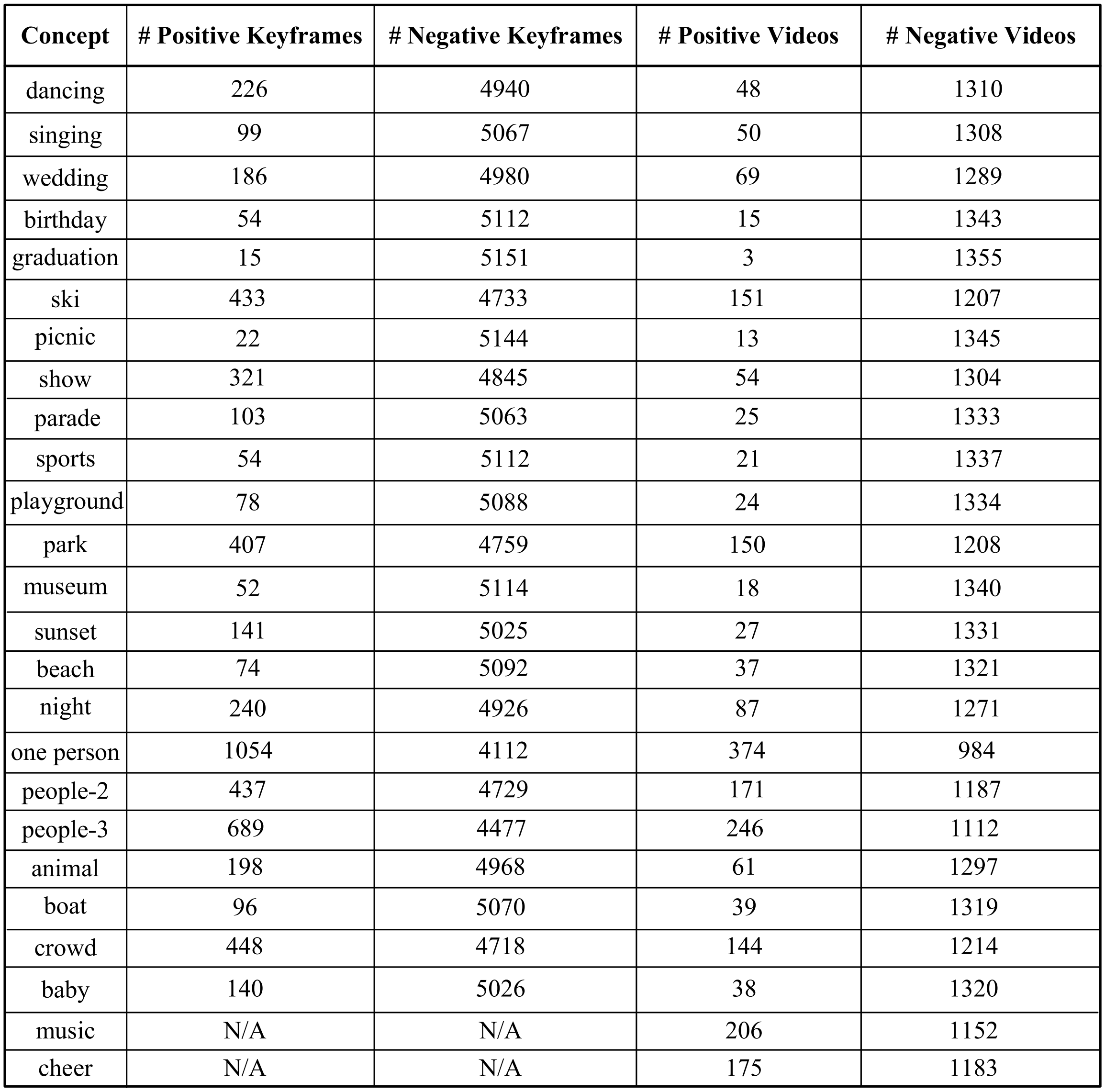
Total Number of Video Clips	1358	
Total Number of Keyframes	5166	
Length of Videos	min	0.1 s
	max	393.1 s
	avg	31.1 s
Resolution	640 x 480 or 320 x 240 (pixels)	
Video Format	MPEG-1	
Bit Rate (Audio + Visual)	min	280 kb/s
	max	19.115 kb/s
	avg	7.99 kb/s
Frame Rate	30 frames / second	
Audio Sampling Rate	44100 Hz	

the video clips. Note for audio-based analysis, typically the soundtrack of the entire video clip is used, rather than just audio signals associated with the keyframes. However, in practice we may extract audio signals just near the keyframe time point in order to combine the local audio cues with the visual cues found near the keyframe time point. To ensure quality of the extracted keyframes, we deliberately insert an initial time offset to the sample schedule. An offset of 1 second is applied at the beginning because the first frames (time = 0) of some video clips are totally black or blurred. In other words, keyframes are extracted at the following time points: 1 s, 11 s, 21 s, 31 s, etc. In addition, to avoid missing important content, if the duration of a video clip is less than 11 s, the final frame of the clip will be included automatically. Although we could have used an automatic keyframe-extracting algorithm, we did not do so because the algorithm has not been fully evaluated and does not always produce consistent results. Using the simple temporal subsampling technique described above at least ensures consistency.

2.7.1.3 Annotation

The concept labels for Kodak's video data set are manually annotated by students at Columbia University. To increase the throughput of the annotation process and ensure good quality of the resulting labels, we employed a multi-tier annotation strategy. For visual-oriented concepts (*e.g.*, activities, occasions, scenes, people, objects), we always ob-

Table 2.3: The number of positive and negative keyframes and video clips in Kodak's consumer video data set.



Concept	# Positive Keyframes	# Negative Keyframes	# Positive Videos	# Negative Videos
dancing	226	4940	48	1310
singing	99	5067	50	1308
wedding	186	4980	69	1289
birthday	54	5112	15	1343
graduation	15	5151	3	1355
ski	433	4733	151	1207
picnic	22	5144	13	1345
show	321	4845	54	1304
parade	103	5063	25	1333
sports	54	5112	21	1337
playground	78	5088	24	1334
park	407	4759	150	1208
museum	52	5114	18	1340
sunset	141	5025	27	1331
beach	74	5092	37	1321
night	240	4926	87	1271
one person	1054	4112	374	984
people-2	437	4729	171	1187
people-3	689	4477	246	1112
animal	198	4968	61	1297
boat	96	5070	39	1319
crowd	448	4718	144	1214
baby	140	5026	38	1320
music	N/A	N/A	206	1152
cheer	N/A	N/A	175	1183

tain annotations of individual keyframes using keyframe-based annotation tools. Such an approach is sufficient for most static concepts. For concepts that involve information in the temporal dimension, we further employ video playback tools to verify the correctness of the label. We do this in an incremental manner; namely, only those keyframes receiving positive labels in the first step are included in the video-based verification process. During the verification step, an annotator plays back each candidate video clip and marks the presence or absence of the concept. Keyframes corresponding to negative videos (where the concept is absence) are corrected as negative. In this case, it is possible that only a subset of keyframes of a video receive positive labels, while the remainder are negative. We use the above incremental procedure to avoid the high workload involved in using video playback to annotate every clip. Based on our experience, the keyframe-based annotation process is much faster than the video-based process. On average, the throughput of the keyframe-based annotation process is about 1-3 seconds per keyframe, while the throughput for the video-based annotation is about 30-60 seconds per video. Finally, for audio-oriented concepts (*e.g.*, “music”, “cheer”, and “singing”), we use the video-based annotation process to label every video clip. Binary labels are assigned for each concept, indicating its presence or absence. Table 2.3 shows the number of positive and negative keyframes and videos for each concept in the ground-truth annotation. Fig. 2.2 shows the example keyframes of different semantic concepts.

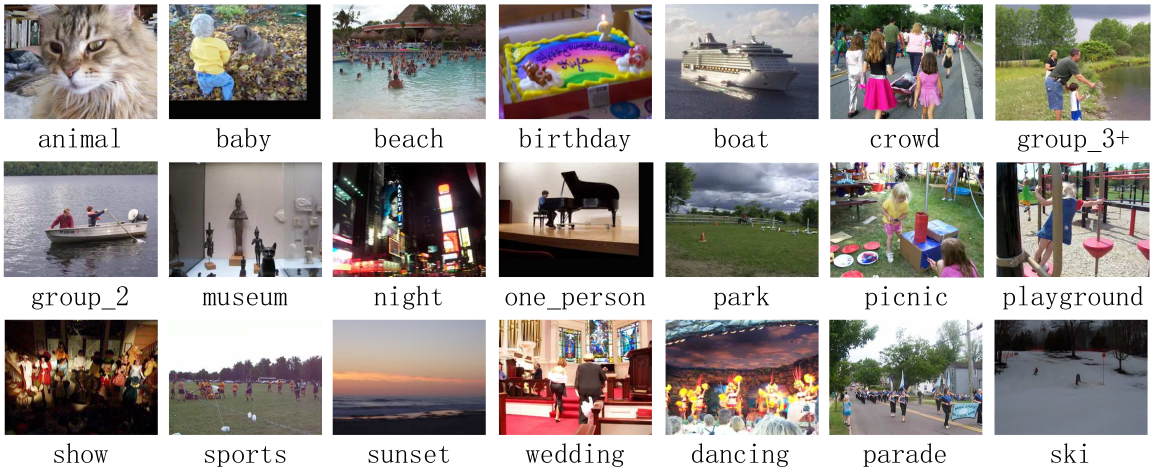


Figure 2.2: Example keyframes for Kodak’s consumer benchmark video set corresponding to different semantic concepts.

2.7.2 TRECVID 2005 Data Set

We also carry out experiments over the challenging TRECVID 2005 news video set [170]. This video set contains 137 (with 80+ hours) broadcast news videos from multilingual broadcast news channels like BBC, NBC, CCTV etc. This is one of the largest and most challenging data sets existing to evaluate concept detection. A common set of concepts has been defined by a *Large-Scale Concept Ontology for Multimedia* (*LSCOM*) developed in [163], which includes 834 concepts jointly selected by news analysts, librarians, and researchers. A subset of these concepts (449) has been annotated through an exhaustive manual process over the entire 2005 TRECVID development set [139]. Large-scale baseline automatic classifiers for LSCOM concepts, such as Columbia374 [248] and MediaMill 491 [243], have been developed and broadly disseminated in the research community. More details about this data set can be found at [170].

In our experiments, the news videos are divided to shots and 60000+ keyframes are extracted from these shots. The keyframes are labeled with 39 concepts from LSCOM-Lite ontology [139], *e.g.*, “car”, “outdoor”, *etc.* In both Kodak’s consumer videos and the TRECVID news videos, an annotated keyframe can belong to different semantic concepts. That is, both are multi-label data sets.

2.7.3 Performance Evaluation Metric

The experiments presented in this thesis use the *Average Precision* (*AP*) and *Mean Average Precision* (*MAP*) as performance measurements. AP is an official TRECVID performance metric, which is related to multi-point average precision value of a precision-recall curve [170]. To calculate AP for a concept C^i, we first rank the test data according to the classification posteriors of concept C^i. Then from top to bottom, the precision after each positive sample is calculated. These precisions are averaged over the total number of positive samples for C^i. AP favors highly ranked positive samples and combines precision and recall values in a balanced way. MAP is calculated by averaging APs across all concepts.

2.7.4 Global Visual Features and Baseline Models

In the thesis we compare most of the proposed algorithms with the following SVM-based baseline method, which has been proven to be state-of-the-art for generic concept detection in general videos [2, 27, 170, 248]. Three types of global visual features are extracted: *Gabor texture* (*GBR*), *Grid Color Moment* (*GCM*), and *Edge Direction Histogram* (*EDH*).

These features have been successfully used for generic video concept detection in [170, 248]. The GBR feature is used to estimate the image properties related to structures and smoothness; GCM approximates the color distribution over different spatial areas; and EDH is used to capture the salient geometric cues like lines. A detailed description of these features can be found in [249].

Using these global visual features, one SVM classifier is trained over each of the three features individually. Then the detection scores from all different SVM classifiers are averaged to generate the baseline visual-based concept detector. The SVMs are implemented using LIBSVM (Version 2.81) [23] with the RBF kernel $k(x_1, x_2) = \exp\{-\gamma||x_1 - x_2||^2\}$. To learn each SVM classifier, we need to determine the parameter setting for both the RBF kernel (*i.e.*, parameter γ) and the SVM model (*i.e.*, parameter C in Section 2.2.2). Here we employ a multi-parameter set model instead of cross-validation so that we can reduce the degradation of performance in the case that the distribution of the validation set is different from the distribution of the test set. Such multi-parameter set model has been shown effective by previous experiments in [24]. Instead of choosing the best parameter set from cross-validation, we average the scores from the SVM models with 25 different pairs of C and γ:

$$\begin{aligned} C &\in \left\{2^0, 2^2, 2^4, 2^6, 2^8\right\} \\ \gamma &\in \left\{2^{k-4}, 2^{k-2}, 2^k, 2^{k+2}, 2^{k+4}\right\} \end{aligned}$$

where $k = ROUND\left(\log_2\left(1/|D_f|\right)\right)$ and $|D_f|$ is the dimensionality of the feature vector based on which the SVM classifier is built ($\gamma = 2^k$ is the recommend parameter in [23]). The multi-parameter set approach is applied to each of the three features mentioned above. Note the scores (*i.e.*, distances to the SVM decision boundary) generated by each SVM are normalized by a standard sigmoid function $\frac{1}{1+e^{-(\cdot)}}$ before averaging.

2.7.5 BoF over Kodak's Consumer Benchmark Videos

As discussed in earlier part of this chapter, the BoF representation generated from unordered local descriptors has become increasingly popular for generic concept detection. State-of-the-art classification results are achieved over large-scale public benchmarks, *e.g.*, TRECVID videos [170] and PASCAL VOC data [182], by generating multiple BoF representations over different local descriptors extracted from various interest keypoints, and by using various matching strategies for kernel construction.

In this subsection, we apply the state-of-the-art BoF-based approaches to Kodak's consumer benchmark video set [137]. Such comparison provides additional information to help analyze consumer videos as well as guide future research to construct efficient consumer concept detectors.

2.7.5.1 Local Descriptors

There have been many research efforts to develop various local descriptors extracted from different interest keypoints for image and video classification. The recent activities in PASCAL VOC and NIST TRECVID have suggested several types of SIFT-based features as well as local interest keypoints [101, 170, 182, 228], which are described below.

- As suggested by Van de Sande *et al.* in [228], the following SIFT-based descriptors can generate reliable performance over both the PASCAL VOC image benchmark data and the TRECVID video benchmark data.
  - The SIFT descriptor proposed by Lowe [138], which is shift-invariant and scale-invariant to light intensity. However, the SIFT descriptor is not invariant to color changes, because the intensity channel is a combination of the R, G and B channels.
  - The HSV-SIFT descriptor developed by Bosch *et al.* in [18]. HSV-SIFT computes SIFT descriptors over all three channels of the HSV color space, which gives a 3x128-dimension feature. The HSV-SIFT descriptor has no invariance properties due to the combination of the HSV channels.
  - The Opponent-SIFT descriptor computed in [228]. It calculates SIFT descriptors over all three channels of the opponent color space. Opponent-SIFT is shift-invariant and scale-invariant to light intensity. However, it is not invariant to color changes.

  - The C-SIFT descriptor proposed in [20]. C-SIFT uses the C-invariant [62] that can be intuitively seen as the normalized opponent color space. C-SIFT is only scale-invariant to light intensity.
  - The rgSIFT descriptor computed in [228]. It calculates SIFT descriptors over the r and g chromaticity components of the normalized RGB color model, which gives 2x128-dimension feature. rgSIFT is also only scale-invariant to light intensity.
  - The Transformed color SIFT descriptor computed in [228]. It calculates SIFT descriptors over all three normalized R, G, and B color channels. The descriptor is invariant to both light intensity and color changes.

  These descriptors are all extracted over the scale-invariant keypoints obtained with the Harris-Laplace point detector on the intensity channel. This detector uses the Harris corner detector to find potential scale-invariant points, and then selects a subset of these points for which the Laplacian-of-Gaussians reaches a maximum over scale.

- As suggested by Chang *et al.* in [25], the intensity-based SIFT descriptors extracted over interest keypoints found by the following detectors are effective for detecting multimedia concepts in the TRECVID video benchmark data: the DoG detector [138]; the Hessian Affine detector [154]; and the MSER detector [152].

The BoF method treats an image as a collection of unordered local descriptors, and then quantizes them into discrete visual words coming from a visual vocabulary. In this subsection, the K-means algorithm is applied to construct a visual vocabulary on top of each type of SIFT-based local descriptors described above (9 visual vocabularies in total), where each visual vocabulary contains 1000 visual codewords.

2.7.5.2 BoF with Different Spatial Layouts

After getting visual vocabularies, the next step of the BoF method is to compute a histogram feature for each image based on each visual vocabulary, where each bin in the histogram corresponds to a visual word in the vocabulary.

Following the approaches in [25, 228], we use four different spatial partitions of images to generate four BoF histograms with each visual vocabulary, as shown in Fig. 2.3. For each visual vocabulary, within each partitioned block, the corresponding local descriptors are mapped to this visual vocabulary to generate a histogram using the soft-weighting strategy proposed in [101]. The final BoF histogram is obtained by concatenating the histograms from different blocks together. So in the end we have in total of 9 × 4 BoF histograms to describe images.

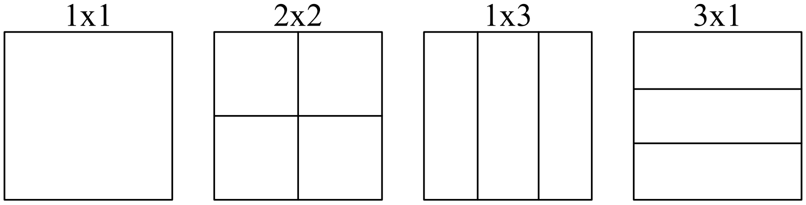


Figure 2.3: Four spatial partitions of images to generate BoF histograms.

2.7.5.3 Kernels and Classification

Using each BoF histogram feature, various kernel matrices can be constructed to use SVM for concept detection. Following the suggestions from [101] as well as considering the experiments in Section 2.7.4, four different RBF kernels are computed using the following four types of distances:

$$k(\mathbf{x}_1, \mathbf{x}_2) = \exp\{-\gamma d(\mathbf{x}_1, \mathbf{x}_2)\} \tag{2.3}$$

where

$$d(\mathbf{x}_1, \mathbf{x}_2) = \begin{cases} \sum_i (x_{1,i} - x_{2,i})^2 \\ \sum_i |x_{1,i} - x_{2,i}| \\ \sum_i \sqrt{|x_{1,i} - x_{2,i}|} \\ \sum_i \frac{(x_{1,i} - x_{2,i})^2}{x_{1,i} + x_{2,i}} \end{cases} \tag{2.4}$$

$x_{1,i}$ and $x_{2,i}$ are the i-th entry of $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively.

In addition, the multi-parameter set approach described in Section 2.7.4 is used. That is, we set γ to five different values (the error control parameter C in SVM is set to be 1 as suggested in [23]):

$$\gamma = \left(\frac{1}{|D_f|}\right)^g, \quad g = \frac{1}{4}, \frac{1}{2}, 1, 2, 4 \tag{2.5}$$

Finally, we have in total of 9 × 4 × 4 × 5 SVM classifiers trained to detect each semantic concept based on the BoF representations.

2.7.5.4 Experimental Results

We compare the above BoF-based classifiers with the SVM classifiers using global image-level visual features. In the global approach, the three visual features, *i.e.*, GBR, GCM, and EDH described in Section 2.7.4, are concatenated into a long feature vector to train one SVM classifier for detecting each semantic concept.

Fig. 2.4 shows the AP and MAP comparison between the global approach and the local approach with SIFT descriptors used in [25] (*i.e.*, SIFT over DoG, Hessian Affine, and MSER interest keypoint detectors). Fig. 2.5 gives the AP and MAP comparison between the global approach and the local approach using various SIFT-based descriptors with the Harris-Laplace interest keypoint detector (*i.e.*, SIFT, C-SIFT, HSV-SIFT, rgSIFT, Opponent-SIFT, and Transformed color SIFT), as suggested in [228]. Finally, Fig. 2.6 shows the comparison between global and combinations of local approaches.

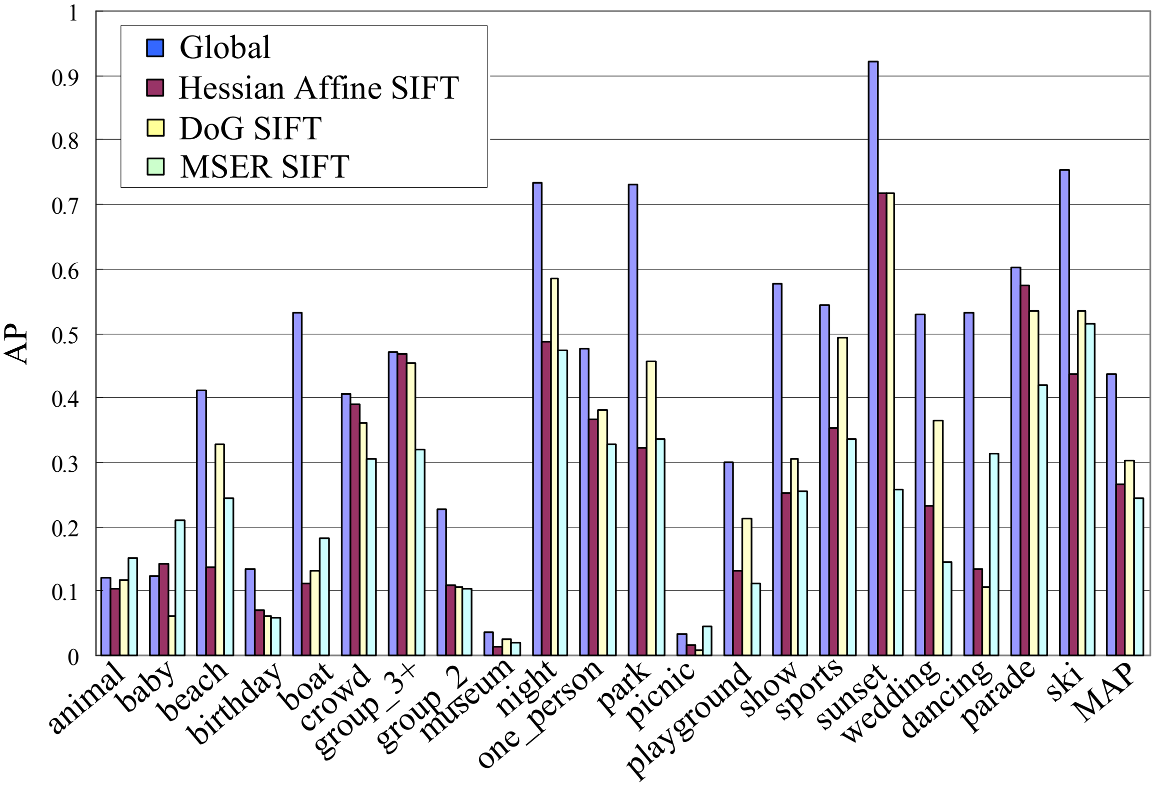


Figure 2.4: Performance comparison: global features versus local SIFT over different interest keypoints.

From these figures, global visual features are quite effective in detecting consumer concepts over Kodak's videos in terms of both accuracy and computation (taking only about 10 seconds to compute per image). In comparison, individual types of local descriptors, although having been proved effective in detecting objects and scenes in other large-scale benchmarks, can not compete with global features in general over this consumer video set. For example, it takes about 2 minutes to compute a SIFT-based descriptor over an image. Specifically, local descriptors can outperform global features over some objects and scenes involving man-made structures, such as "animal", "baby", "boat", "museum", and "playground", while global features win over natural scenes like "night", "park", "ski", and "sunset" where the visual appearance of global background is important. On average, color-based local descriptors (in Fig. 2.5) work better than intensity-based ones (in Fig. 2.4).

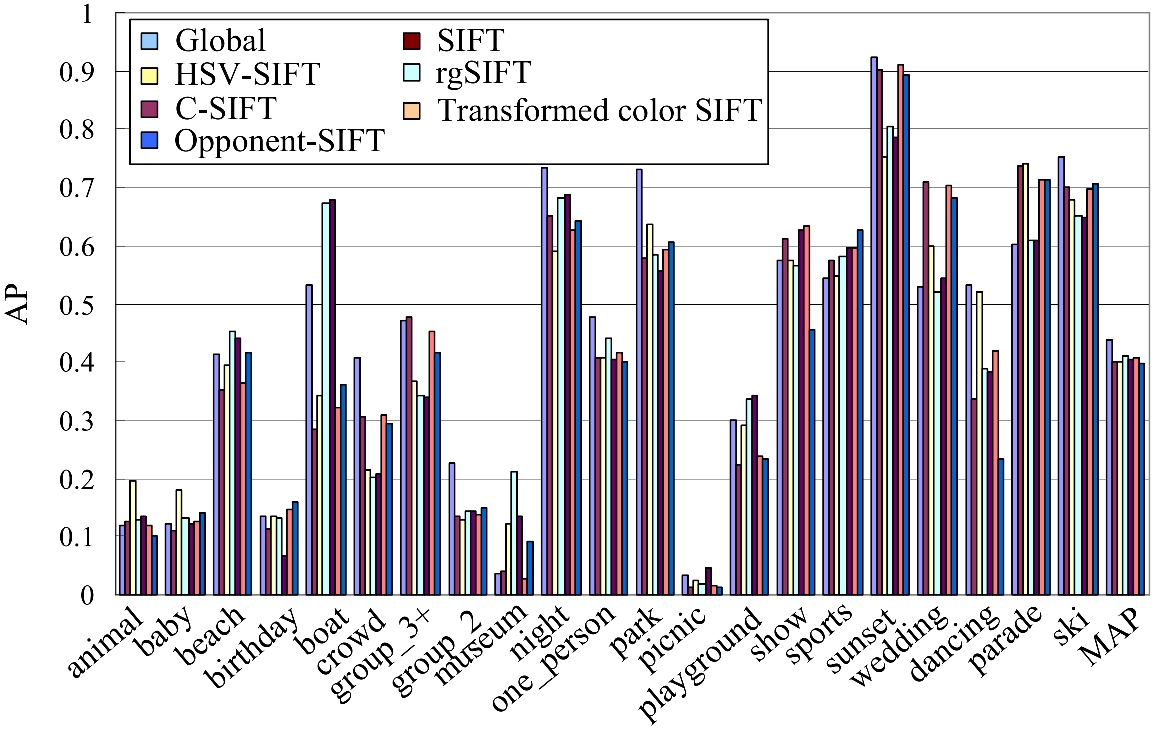


Figure 2.5: Performance comparison: global features versus different SIFT-based descriptors over the Harris-Laplace detector.

In Fig. 2.6 when we aggregate classifiers learned over various local descriptors, we can get better detection precision than using individual local descriptors. For example, the "SIFT+ColorSIFT (Harris-Laplace)" approach has similar MAP with the global method, and the "SIFT (DoG+Hessian-Affine+MSER)" approach, although still performs worse than the global method, gives better MAP than individual SIFT features over DoG, Hessian-Affine, or MSER. Finally when we combine all various local descriptors together (the "Local All" approach), the BoF-based local method outperforms the global method by a small margin. Specifically, the "Local All" method wins over 10 concepts compared with the global method, *i.e.*, "animal", "baby", "beach", "birthday", "dancing", "group_3+", "museum", "show", "sports", and "wedding". Most of these concepts are object/scene-oriented.

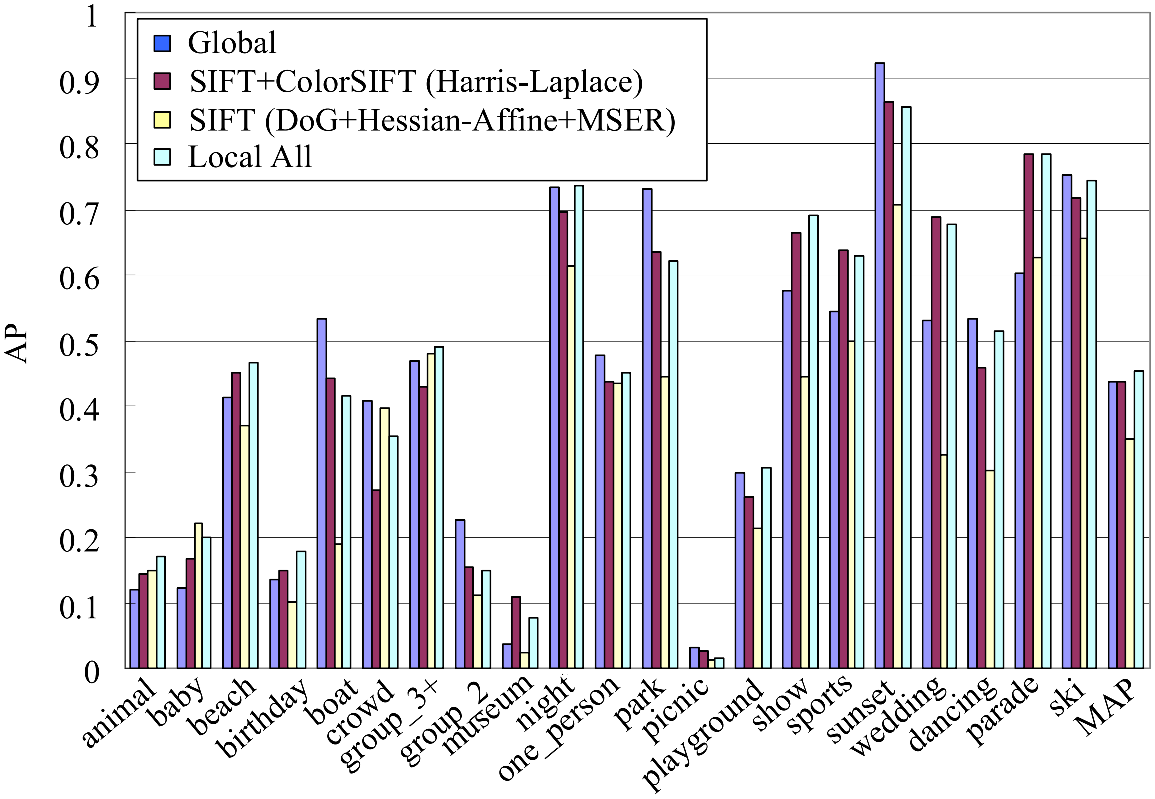


Figure 2.6: Performance comparison: global features versus local descriptors. "SIFT (DoG+Hessian-Affine+MSER)" combines classification results using the three SIFT descriptors over DoG, Hessian Affine, and MSER interest keypoints. "SIFT+ColorSIFT (Harris-Laplace)" combines classification results using SIFT, C-SIFT, HSV-SIFT, rgSIFT, Opponent SIFT, and Transformed color SIFT over Harris-Laplace interest keypoints. "Local All" combines all different SIFT-based descriptors together.

The experimental results obtained in this experiment are different from those in some other literatures such as [25, 228] where local descriptors generally outperform global visual features in detecting objects in images from other domains (*e.g.*, Internet and Broadcast). This is because of the challenging conditions of detecting generic concepts in the consumer domain. The visual content of general consumer videos can be very diverse, with very little repetition of objects or scenes in different videos, even those from the same concept class. For example, Vivian's birthday party may be totally different from John's birthday party in terms of human subjects, room setting, *etc.* In such a case, it is hard to leverage the advantage of local descriptors like SIFT for local point matching/registration. This is quite different from other benchmarks like PASCAL VOC images [182] or TRECVID videos [170] where similar objects or scenes repetitively occur in many different images/videos. In addition, many general consumer videos have various problems such as out of focus, ill-posed camera, very small/large objects, *etc.* This is quite different from PASCAL VOC images or TRECVID videos. In such situations, neither global nor local descriptors can really describe the object of interest, and in comparison global features can capture robust background setting and may perform better. Fig. 2.7 gives some examples to illustrate the challenging conditions in general Kodak's consumer videos.

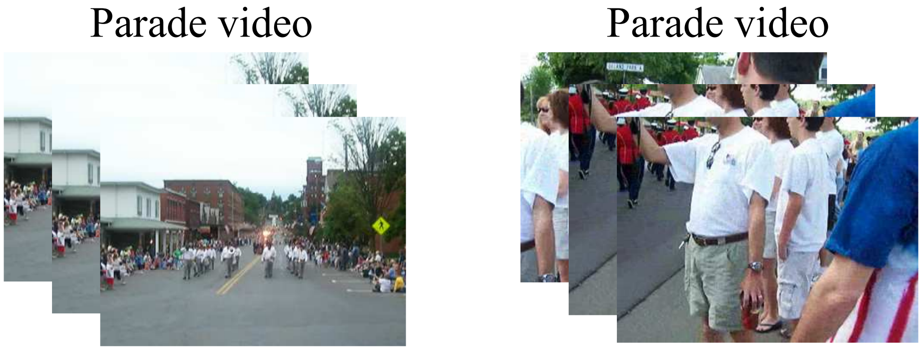


Figure 2.7: Example of Kodak's consumer videos illustrating the challenging conditions for concept detection: little repetition of objects or scenes; very small/large objects; ill-posed camera; *etc.*

Chapter 3

Multi-Concept: Context-Based Concept Fusion

3.1 Introduction

As discussed in Section 1.2.1, in the real-world scenario, semantic concepts usually do not occur in isolation. For example, knowing the contextual information (*e.g.*, “outdoor”) of an image is expected to help detection of other concepts (*e.g.*, “car”). In this chapter we study new methods of multi-concept learning where the inter-conceptual relationships are used to enhance classification of each individual concept.

In Section 2.6 we have introduced previous works utilizing context to help object or region recognition or to enhance image and video concept classification. The two-layer CBCF framework is particularly useful for detecting generic video concepts. In the first layer of CBCF, independent concept detectors are applied to estimate posteriors of concept labels in a given image. In the second layer detection results of individual concepts are updated by taking into account detection confidences of other concepts and the inter-conceptual relationships.

Several CBCF methods have been proposed, including the Multinet method [162], the Bayesian network method [179], the SVM-based *Discriminative Model Fusion* (*DMF*) method [214], *etc.* However, results reported so far [2, 215] have indicated that not all concepts benefit from CBCF learning. The lack of consistent performance gain could be attributed to several reasons: 1) insufficient data for learning reliable relations among concepts, and 2) unreliable individual concept detectors in the first layer.

In this chapter we model the inter-conceptual relationships by a CRF [114] (as shown in Fig. 3.1). For each image I, CRF takes as inputs the detection results, $\mathbf{h}_I = [\hat{P}(y_I^1 = 1|I), \ldots, \hat{P}(y_I^M = 1|I)]^T$ (y_I^i is the label for concept C^i in image I), from M independent concept detectors, and produces updated marginal probabilities $P(y_I^i\!=\!1|I)$ of each concept C^i. CRF directly models the conditional distribution $P(\mathbf{y}_I|\mathbf{h}_I)$ of class label $\mathbf{y}_I$ given input observation $\mathbf{h}_I$, while the generative methods (*e.g.*, Multinet [162]) model the joint distribution $P(\mathbf{y}_I, \mathbf{h}_I)$. When the training set is limited, the discriminative CRF can better use the sample resources to model the distribution relevant to the discriminative concept detection task than the generative approach.

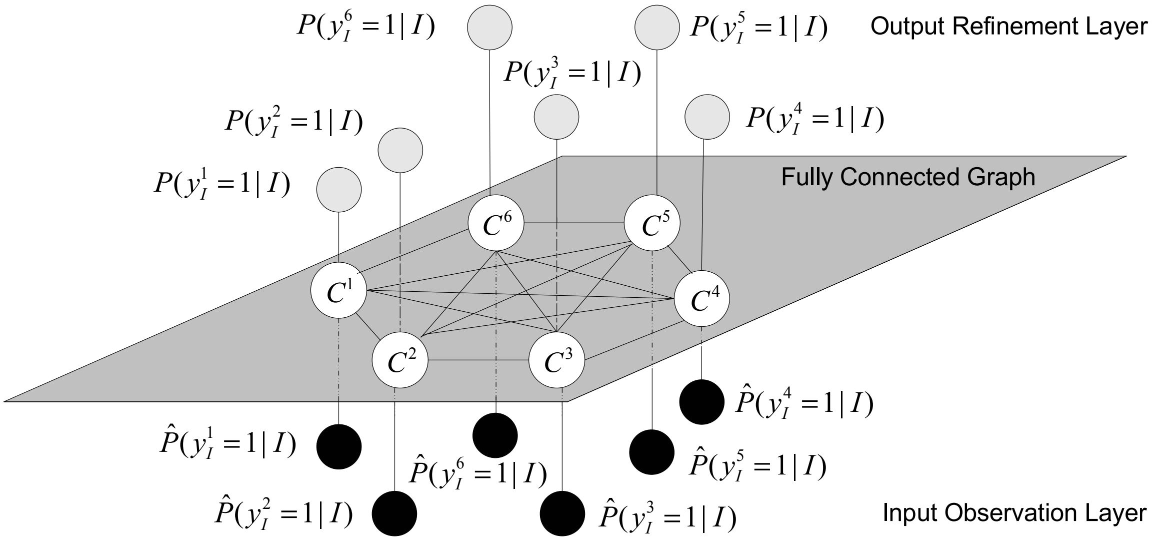


Figure 3.1: The CRF that models relationships among M concepts $C^1, \ldots, C^M$. Concepts are related to each other, so the graph is full connected. The CRF takes as input (black nodes) detection results, $\hat{P}(y_I^1 = 1|I), \ldots, \hat{P}(y_I^M\!=\!1|I)$, from M independent concept detectors, and produces updated marginal probabilities $P(y_I^i\!=\!1|I)$ (gray nodes) of each C^i. $\mathbf{y}_I\!=\![y_I^1, \ldots, y_I^M]$ is the vector of concept labels.

With the two-layer CBCF framework and the CRF context modeling, we develop multi-concept learning algorithms to enhance semantic concept detection performance. Our contributions can be summarized into three aspects.

- First, to avoid the difficulty of designing compatibility potentials in CRF, a discriminative cost function aiming at class separation is directly optimized. This is in contrast to the previous graph-based CBCF methods [162, 247] that use co-occurrence statistics of concepts to construct compatibility potentials. With limited training data, the unreliable co-occurrence statistics will make performance suffer. To optimized the discriminative cost function, we adopt the Boosted CRF framework [222], and use the Real AdaBoost algorithm [59] to iteratively improve concept detection. SVM is used as the weak learner for boosting because of its state-of-the-art performance in detecting semantic concepts over general videos, *e.g.*, consumer videos [24, 137] or TRECVID videos [2, 170]. The proposed algorithm is called Boosted CRF–Concept Fusion. As will be shown in Section 3.2 the traditional DMF [214] corresponds to the initial stage of BCRF-CF. We will also show that the extended iterative steps introduced in our method can further improve the performance.

- Second, to address the problem of inconsistent performance gain by CBCF, we propose a simple but effective criterion to predict which concepts will benefit from CBCF, based on both information theoretic and heuristic rules. This criterion takes into consideration both the strength of relationships between a concept and its neighborhood and the robustness of detections of this neighborhood. The accurate prediction scheme allows us to use CBCF in practice, applying it only when it is likely to be effective.

- Third, based on the BCRF-CF algorithm, we further study the interesting "20 questions" problem in semantic concept detection. We propose the Active CBCF algorithm to effectively exploit semantic context and active user input to enhance concept detection. We develop a concept selection method by considering the mutual information between concepts (representing how much a concept can help), and the accuracy of the concept detector as well as its confidence of prediction over a specific image (representing how much a concept needs help). A set of image-varying concepts are actively selected for the user to label so that user's annotation can optimally help detect all concepts overall.

In our experiments, we evaluate the proposed BCRF-CF and Active CBCF algorithms over the TRECVID 2005 development set. Out of the 39 LSCOM-Lite concepts, 26 are automatically chosen to use the context-based fusion. Among these 26 concepts, 21 concepts get noticeable performance gains by using BCRF-CF, and 24 concepts get noticeable performance gains by using Active CBCF. In other words, the accuracy of prediction is 81% for BCRF-CF and 92% for Active CBCF, respectively. Also, the Active CBCF approach can consistently outperform the passive CBCF method where randomly selected concepts are provided for the user to label.

3.2 Boosted CRF Concept Fusion

As mentioned above, in the CBCF scenario, for each image I the observations (system inputs) are the posteriors $\mathbf{h}_I = [h_I^1, \ldots, h_I^M]^T$, $h_I^i = \hat{P}(y_I^i = 1|I)$, namely, the classification scores of independent concept detectors. Our goal is to feed these inputs into an inferencing model to get improved posterior probability $P(\mathbf{y}_I|I)$ by taking into account inter-conceptual relationships.

The posterior $P(\mathbf{y}_I|\mathbf{h}_I)$ can be modeled by a CRF [114] as:

$$P(\mathbf{y}_I|\mathbf{h}_I) = \frac{1}{Z} \exp \left\{ \sum_{i=1}^{M} \phi_i(y_I^i, \mathbf{h}_I) + \sum_{i=1}^{M} \sum_{j=1, j\neq i}^{M} \psi_{ij}(y_I^i, y_I^j, \mathbf{h}_I) \right\}$$

Z is a normalizing constant; $\phi_i(y_I^i, \mathbf{h}_I)$ and $\psi_{ij}(y_I^i, y_I^j, \mathbf{h}_I)$ are the local and compatibility potentials respectively. One issue of CRF modeling is the design of potential functions. $\phi_i(y_I^i, \mathbf{h}_I)$ is a local decision term that influences the posteriors of concept C^i independent of its neighbors. Compatibility potentials $\psi_{ij}(y_I^i, y_I^j, \mathbf{h}_I)$ are generally used to specify constraints for relationships between pairs of nodes, *e.g.*, spatially smoothing constraints in image segmentation [114]. In [247], the same CRF-based CBCF framework is used to refine concept detection, where co-occurrence statistics of different concepts are used to construct compatibility potentials. However, the co-occurrence statistics can be unreliable when we only have limited training data, and it is difficulty to design relationships among concept nodes other than co-occurrence statistics. Different from [247], to avoid designing compatibility potentials in this work we incorporate the Boosted CRF framework proposed in [222] that directly optimizes a discriminative cost function based on CRF. In the next subsections we will introduce the Boosted CRF framework [222], followed by our BCRF-CF algorithm.

3.2.1 Boosted CRF

After the inference with CRF the belief b_I^i on each node C^i is used to approximate the posterior: $P(y_I^i = \pm 1|I) \approx b_I^i(\pm 1)$. The aim of CRF modeling is to minimize the total loss $\mathcal{L}$ for all concepts over all training data:

$$\mathcal{L} = -\prod_{(I,\mathbf{y}_I)\in\mathcal{D}}\prod_{i=1}^{M} b_I^i(+1)^{(1+y_I^i)/2} b_I^i(-1)^{(1-y_I^i)/2} \tag{3.1}$$

If we assume that the local potential has the following form:

$$\phi_i(y_I^i, \mathbf{h}_I) = [e^{F_I^i/2}; e^{-F_I^i/2}]$$

where F_I^i is a discriminant function (*e.g.*, a logistic regression stump) of input $\mathbf{h}_I = [h_I^1, \ldots, h_I^M]^T$, then we have the following belief [222]:

$$b_I^i(+1) = \frac{1}{1 + e^{-\left(F_I^i + G_I^i\right)}} \tag{3.2}$$

And then the logarithm of Eqn. (3.1) can be written as:

$$\log \mathcal{L} = \sum_{(I,\mathbf{y}_I)\in\mathcal{D}}\sum_{i=1}^{M} \log\left[1 + e^{-y_I^i(F_I^i + G_I^i)}\right] = \sum_{i=1}^{M} \log \tilde{\mathcal{L}}_i \tag{3.3}$$

where

$$\log\tilde{\mathcal{L}}_i = \sum_{(I,\mathbf{y}_I)\in\mathcal{D}} \log[1 + e^{-y_I^i(F_I^i + G_I^i)}]$$

G_I^i is a discriminant function of belief $\mathbf{b}_I^i = [b_I^1(+1), \ldots, b_I^{i-1}(+1), b_I^{i+1}(+1), \ldots, b_I^M(+1)]^T$. Note the belief term b_I^i is now included in G_I^i, and thus G_I^i can be considered as contextual belief of b_I^i. The derivation is straightforward: when $y_I^i = 1$, $\log[1 + e^{-y_I^i(F_I^i + G_I^i)}] = \log b_I^i(+1)$, and when $y_I^i = -1$, since $b_I^i(-1) = 1 - b_I^i(+1) = 1/[1 + e^{\left(F_I^i + G_I^i\right)}]$, $\log[1 + e^{-y_I^i(F_I^i + G_I^i)}] = \log b_I^i(-1)$.

The loss function $\log \mathcal{L}$ described in Eqn. (3.3) can be iteratively minimized by the LogitBoost algorithm [59]. That is, F_I^i and G_I^i are represented as additive models: $F_I^i(T) =$

$\sum_{t=1}^{T} f_I^i(t)$, $G_I^i(T) = \sum_{t=1}^{T} g_I^i(t)$. $f_I^i(t)$ and $g_I^i(t)$ are two classifiers learned during each boosting iteration, which are combined to generate the weak learner for boosting. $f_I^i(t)$ is the logistic regression stump over input $\mathbf{h}_I = [h_I^1, \ldots, h_I^M]^T$ and $g_I^i(t)$ is the logistic regression stump over the current beliefs $\mathbf{b}_I^i(t) = [b_I^1(+1,t), \ldots, b_I^{i-1}(+1,t), b_I^{i+1}(+1,t), \ldots, b_I^M(+1,t)]^T$.

3.2.2 Boosted CRF–Concept Fusion

Motivated by the work of [222] we avoid designing the compatibility potentials (that are very difficult to obtain in our problem) by optimizing the discriminative cost function Eqn. (3.3) with a BCRF-CF algorithm. Mainly two modifications are made to apply the Boosted CRF framework.

(1) SVM classifiers are used instead of logistic regression stumps. SVM performs reasonably well in semantic concept detection for general videos [2, 24, 215], because of its good generalization ability with limited training data. Also the training data is usually highly biased (the positive training samples for a concept is much less than the negative ones), and SVM is more suitable for such unbalanced situations than logistic regression because of the use of support vectors. There is another reason that logistic regression may not work well in our CBCF problem. As discussed in [222] linear regression would work well when the CRF graph was densely connected, *i.e.*, there were a large number of nodes (image pixels in [222]). But the number of nodes (concepts) in our graph is small. Thus the linear approximation of the discriminant function G_I^i made in [222] may be not valid anymore [1]. More complex function should be assumed for G_I^i, *e.g.*, the nonlinear discriminant function from kernel-based SVM.

(2) The Real AdaBoost algorithm is used instead of LogitBoost. LogitBoost uses logistic regression stumps as weak learners, and we need to adopt the general Real AdaBoost algorithm that can use any weak learner. The solution of minimizing $\log \tilde{\mathcal{L}}$ coincides

[1] In dense graphs with weak and symmetrical connections, the message of any single node passing to another node is not informative. In such a case term G_I^i (the influence of other nodes to a specific node) may be simplified as a linear combination of their beliefs [222].

with the solution of minimizing the following cost function Q [59]:

$$Q = \sum_{i=1}^{M} \sum_{(I,\mathbf{y}_I)\in\mathcal{D}} e^{-y_I^i \Gamma_I^i}, \qquad \Gamma_I^i = \frac{F_I^i + G_I^i}{2} \tag{3.4}$$

Eqn. 3.4 is exactly the cost function of Real AdaBoost [59], where we have the following additive model:

$$\begin{aligned} \Gamma_I^i(T) &= \sum_{t=1}^{T} \gamma_I^i(t) \\ \gamma_I^i(t) &= \frac{f_I^i(t) + g_I^i(t)}{2} \end{aligned}$$

That is, the Real AdaBoost algorithm can be used to minimize Eqn. 3.4 iteratively. During each boosting iteration t, $f_I^i(t)$ is the discriminant function generated by an SVM learned over the input $\mathbf{h}_I$; $g_I^i(t)$ is the discriminant function generated by an SVM built on top of the current beliefs $\mathbf{b}_I^i(t)$; and $\gamma_I^i(t)$ is the overall discriminant function, obtained by averaging $f_I^i(t)$ and $g_I^i(t)$.

The detailed BCRF-CF algorithm is given in Fig. 3.2. The initial step of BCRF-CF is exactly the DMF approach proposed in [214]. As we will see in the experiments, this DMF method can get performance improvements over some concepts while degrading the performance over many other concepts, and our boosting process can avoid this problem and achieve more consistent improvements.

3.3 Which Concepts to Update

Not all concepts can benefit from the CBCF learning. For example, the experiment in [2] has shown that only 8 out of the 17 concepts they evaluated gained performance. Although experiments in [215] showed improvements on 80 out of 101 concepts, our baseline independent detectors are stronger than theirs. For example, our baseline detector get 61% AP on "car", while theirs get only 25%. Starting with stronger baseline independent detectors, it is more difficult to achieve significant improvement using CBCF. Therefore, it is worthwhile to study the problem – when will CBCF help and which target concepts should we focus on when applying CBCF?

```
Input: training set $\mathcal{D}$, and posteriors $\mathbf{h}_I=[h_I^1,\dots,h_I^M]^T$ from independent concept detectors, $h_I^i=\hat{P}(y_I^i=1|I)$. $I$ is an input image and $i$ is the index of concepts.

• Initialization:

    For each concept $C^i$:

    – Train an SVM classifier $H^i(0)$ based on $\mathbf{h}_I$, and compute $p^0(y_I^i=1|I)$ based on the classifier score.
    – Set $\gamma_I^i(0) \leftarrow \frac{1}{2}\log\frac{p^0(y_I^i=1|I)}{1-p^0(y_I^i=1|I)}$; $\Gamma_I^i(0)=\gamma_I^i(0)$; $b_I^i(+1,0)=\frac{1}{1+e^{-\Gamma_I^i(0)}}$;
      $w_I^i(0)=\exp[-y_I^i\gamma_I^i(0)]$.

• For $t=1,\dots,T$:

    For each concept $C^i$:

    – Form a new training data set $\tilde{\mathcal{D}}$ with size $|\mathcal{D}|$ by sampling the original set $\mathcal{D}$ according to sample weights $w_I^i(t-1)$.
    – Train SVM classifiers $H_f^i(t)$ and $H_g^i(t)$ with $\tilde{\mathcal{D}}$ as training data, $\mathbf{h}_I$ and $\mathbf{b}_I^i(t-1)=[b_I^1(+1,t-1),\dots,b_I^{i-1}(+1,t-1),b_I^{i+1}(+1,t-1),\dots,b_I^M(+1,t-1)]^T$ as features, respectively. Get the corresponding class probability estimation $p_f^t(y_I^i=1|I)$ and $p_g^t(y_I^i=1|I)$, and also get:

          $$p^t(y_I^i=1|I)=\frac{p_f^t(y_I^i=1|I)+p_g^t(y_I^i=1|I)}{2}$$

    – $f_I^i(t)+g_I^i(t) \leftarrow \log\frac{p^t(y_I^i=1|I)}{1-p^t(y_I^i=1|I)}$; $\gamma_I^i(t)=\frac{f_I^i(t)+g_I^i(t)}{2}$.
    – Update $\Gamma_I^i(t)=\Gamma_I^i(t-1)+\gamma_I^i(t)$; $b_I^i(+1,t)=1/(1+e^{-2\Gamma_I^i(t)})$;
      $w_I^i(t) \leftarrow w_I^i(t-1)\ e^{(-y_I^i\gamma_I^i(t))}$, and re-normalize to make $\sum_{(I,\mathbf{y}_I)\in\mathcal{D}} w_I^i(t)=1$.
```

Figure 3.2: The BCRF-CF algorithm.

Intuitively, two reasons may cause performance deterioration using CBCF: (1) the concept has weak relations with other concepts; (2) the related concepts have poor detectors that tend to generate unreliable information. In the first case, ideally such a concept will not influence or be affected by other concepts. However, in reality the limited training resources may be wasted on learning context-based detector for such a concept. For example, such an independent concept will contribute a noisy dimension to influence learning the g_I^i term in the BCRF-CF algorithm. This suggests an intuitive criterion: a concept C^i should use CBCF learning when C^i is strongly related to other concepts, and the performance of detectors of the related concepts is in general strong. In other words, if a concept has a strong independent detector and poor neighborhood, it is not a good candidate for using CBCF. The relationship between C^i and C^j can be measured by their mutual information $I(C^i;C^j)$, which reflects how much information one concept can get from another. The detection error E^i of the detector for C^i estimates its robustness. It can be approximated by evaluation over a separate validation data set. Then our criterion for applying CBCF learning to concept C^i is:

$$E^i > \lambda \ \text{ or } \ \frac{\sum_{j:j\in\mathcal{N}_i} I(C^i;C^j)E^j}{\sum_{j:j\in\mathcal{N}_i} I(C^i;C^j)} < \beta \tag{3.5}$$

The statistical co-occurrence of ground-truth labels of concepts C^i and C^j in the training set is calculated to approximate the probability $P(C^i, C^j)$, based on which mutual information is computed. Note that concept co-occurrence is a very rough approximation of $P(C^i, C^j)$, especially with limited data samples. Actually this usually makes the statistical inferencing methods [162] suffer because during inference the error from inaccurate approximation will accumulate. However this approximation may be sufficient for the simple concept prediction criterion in Eqn. (3.5), since the individual classifier learning does not depend on such approximation.

3.4 Experiments: BCRF-CF

Experiments are carried out on the TRECVID 2005 development set [26, 170], which contains 137 broadcast news videos and has been labeled with 39 concepts from the LSCOM-Lite ontology [164] (see Section 2.7.2 for details). It is separated into two training sets $\mathcal{T}_1$, $\mathcal{T}_2$ and one test set, as shown in Table 3.1. The independent concept detectors used here are the baseline detectors describe in Section 2.7.4, *i.e.*, the SVM-based classifiers over simple image features such as GCM and GBR, extracted from the keyframes of videos. Outputs of SVMs are converted into probabilities through a standard sigmoid function $1/(1+e^{-(\cdot)})$.

Table 3.1: The data sets for experiments to evaluate BCRF-CF.

Name	Size	Usage
training set $\mathcal{T}_1$	41837 keyframes	train independent detectors
training set $\mathcal{T}_2$	13547 keyframes	train Boosted CRF model
test set	6507 keyframes	evaluation

3.4.1 Performance Evaluation

First, we empirically set $\lambda = 0.95$, $\eta = 0$, $\beta = 0.7$ in the concept prediction criterion Eqn. (3.5). These parameters are determined by 3-fold cross validation over training set $\mathcal{T}_2$. A number of 26 semantic concepts are automatically selected (shown in Fig. 3.3) to use the BCRF-CF method. Fig. 3.3 gives the MAP (defined in Section 2.7.3) comparison of BCRF-CF, DMF, and original independent detectors, through 10 iterations over the selected 26 concepts. The figure indicates that both CBCF methods, *i.e.*, BCRF-CF and DMF, improve the overall MAP. DMF corresponds to the initial stage of our BCRF-CF and can achieve 4.2% performance gain. Our BCRF-CF can further enhance the initial DMF performance in successive boosting iterations with 2.6% MAP gain after 10 iterations. The performance improvement in MAP obtained by our BCRF-CF is 6.8% compared with the baseline independent detectors. As we can see in detailed AP comparisons later, due to the challenging conditions of this data set, concept detectors generally perform unevenly over different concepts (from less than 10% AP over "corporate leader" to over 90% AP over "person"). The overall MAP is dominated by some concepts with high APs. A 6.8% overall MAP gain as well as the fact that 81% concepts can get noticeable performance improvement can be considered as significant.

Fig. 3.4 gives individual AP of our BCRF-CF, the DMF and the independent detectors (after 10 iterations) over the selected 26 concepts. DMF obtains performance improvements over 19 concepts, while degrading detection results on the other 7 concepts. The performance deterioration over several concepts are severe, *e.g.*, 8.1% in "vegetation" and 23.6%

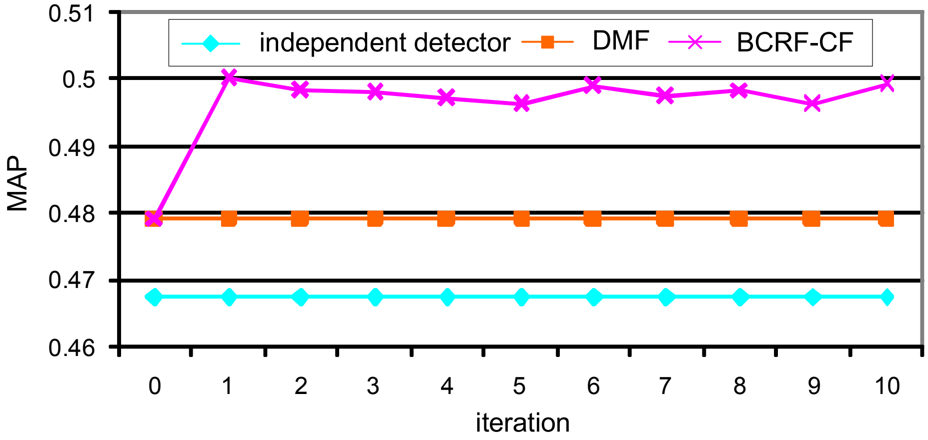


Figure 3.3: The MAP of BCRF-CF, DMF, and independent concept detectors. The results are averaged over 26 selected concepts.

in “walking-running”. Our BCRF-CF algorithm can achieve performance improvement over 21 concepts and avoid significant performance degradation seen in the DMF results. As an example, Fig. 3.5 shows the AP of BCRF-CF over “vegetation” and “walking-running” respectively, evolving through 10 iterations. The figures indicate that BCRF-CF can further improve the performance of DMF. Compared with independent detectors, significant AP improvement is achieved over several concepts by BCRF-CF, *e.g.*, 1221% over “office”.

3.4.2 Evaluation with Different Parameters

Here we evaluate the performance of detection with different β in Eqn. (3.5), since we find that β is the most sensitive parameter for concept prediction. By varying β different concepts are selected to use BCRF-CF. Here we define the *precision* of the concept selection as the percentage of selected concepts that actually have performance gain after applying BCRF-CF. Table 3.2 shows the precision and MAP gain of BCRF-CF after 10 iterations over the selected concepts with $\beta = 1, 0.7, 0.4$ respectively. Such results are promising and can be used to achieve the highest performance-cost ratio. If we don’t do concept selection (*i.e.*, all concepts are included), only 59% of the concepts are improved. With concept selection, computational resources can be allocated to enhance concepts that have good chance to gain improvement. Note that from Fig. 3.3 and Fig. 3.5, the BCRF-CF

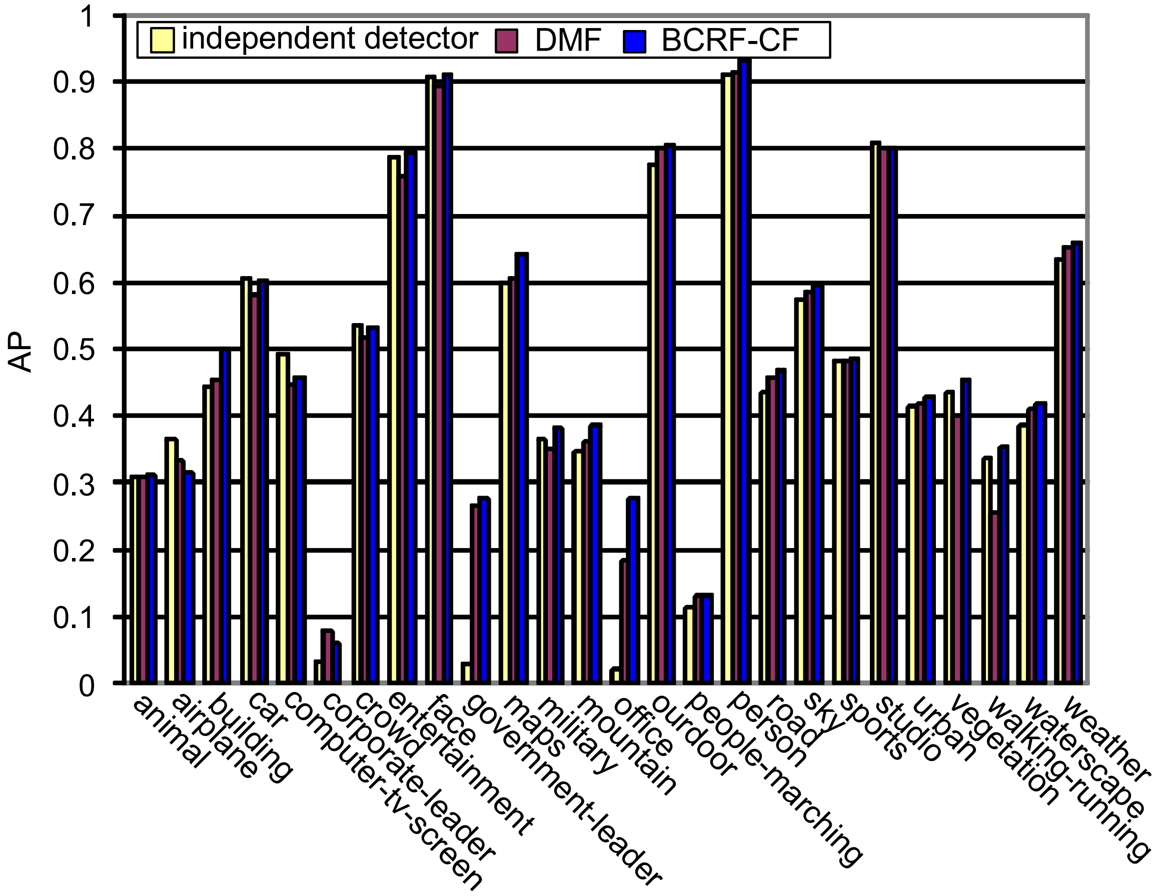
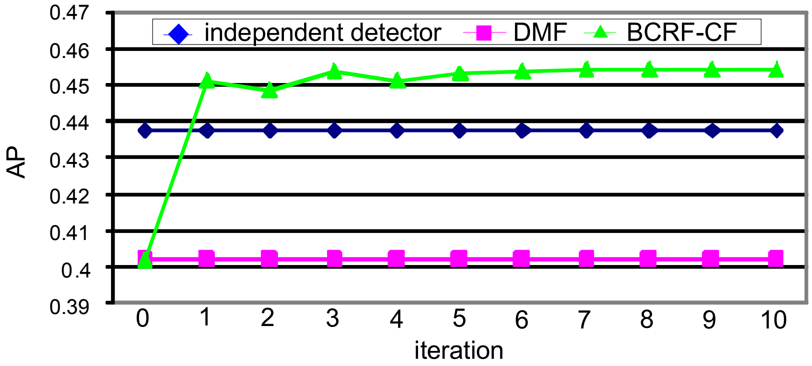


Figure 3.4: The individual AP comparison of our BCRF-CF, DMF, and the baseline independent detector.

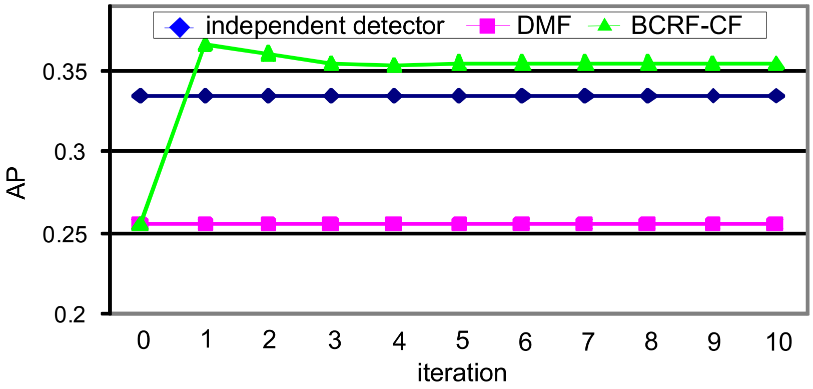
algorithm converges quick to a local minimum in general within the first 2 or 3 iterations. This phenomenon indicates the potential overfitting problem over this data set, which will be seen again in the next chapter (Section 4.4.2). This may be attributed to the small positive training set for many concepts, *e.g.*, 254 positive samples out of 41847 training samples for "boat-ship".

Table 3.2: Performance of detection with different β.

β	precision	MAP gain	# of selected concepts
1	59%	2.2%	39
0.7	81%	6.8%	26
0.4	91%	10.4%	11



(a) Result for “vegetation”



(b) Result for “walking-running”

Figure 3.5: Examples of performance evolution.

3.5 20 Questions Problem: Active CBCF

In this section, we consider an interesting active learning scenario, where we use the semantic context and active user input to enhance concept detection. We propose a new fusion paradigm, called Active CBCF, to effectively exploit the contextual relations among concepts. We observe that in several applications, users may be willing to be involved in the annotation process. That is, given an input image the user may be willing to answer a few simple questions such as annotating whether the image contains some specific concepts. The human annotated concepts are then used to help infer and improve detection of other concepts. Human assisted annotation is not new in the literature. But a new, interesting question arises in the active paradigm – if a user is to annotate only a very small number of concepts *e.g.*, 1∼3 concepts, which concepts should we ask him/her to annotate? We propose an active system that adaptively selects the best concepts, different for each image, for user annotation. We call this problem the "20 Questions Problem" for semantic concept detection. The goal is to be "smart" in asking the right questions for each image so that the overall accuracy in predicting all concepts can be optimized. In contrast, conventional methods are passive. Users are asked to annotate all concepts or a subset of arbitrarily chosen concepts.

Based on the statistical principles, our method considers the mutual information between concepts, the estimated performance of individual concept detectors, and the confidence of detection of each concept for a given image. Experiments over TRECVID 2005 data show that our Active CBCF approach can consistently outperform the passive CBCF method where randomly selected concepts are provided for the user to label. For example, when one concept is labeled by the user over each image, Active CBCF achieves 6.2% improvement in terms of MAP compared with passive CBCF.

3.5.1 The Active CBCF approach

The work flow of the Active CBCF method is illustrated in Fig. 3.6. In Active CBCF the user is willing to label a small number of concepts, and the labeled information is utilized to help detection of other concepts as follows. For a test image I, the user labels a set of n concepts $\mathbf{C}^*$=$\{C^{*1},\ldots,C^{*n}\}$. The labeled ground truth $\hat{P}_g(y_I^{*1}=1|I),\ldots,\hat{P}_g(y_I^{*n}=1|I)$ are used to replace the corresponding hypotheses $\hat{P}(y_I^{*1}=1|I),\ldots,\hat{P}(y_I^{*n}=1|I)$ from initial individual detectors to generate a mixed feature vector. In the mixed feature vector, the entries corresponding to the labeled concepts are the labeled ground truth from user, and the other entries are still the estimated hypotheses from individual detectors. This mixed feature vector is provided to the CBCF algorithm to get a new classification result $P_{ACBCF}(y_I^i=1|I)$. Since the labeled ground truth provides accurate inputs instead of the machine-predicted hypotheses, $P_{ACBCF}(y_I^i=1|I)$ is expected to be better than the original $P(y_I^i=1|I)$ learned from the original inputs. Note that in this version of our work, user provides binary labels, *i.e.*, $\hat{P}_g(y_I^{*i}=1|I)$=1 or 0. Soft scores provided by the user can also be adopted here.

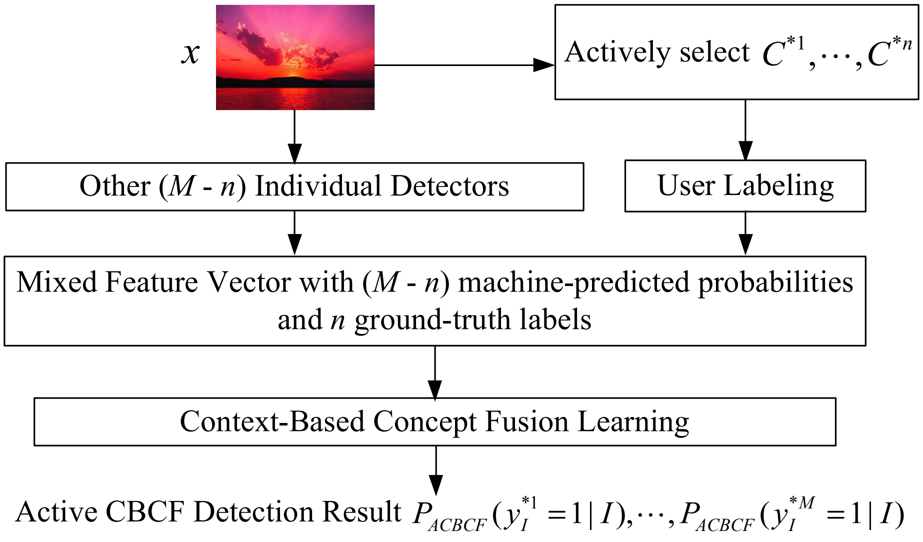

x
Actively select $C^{*1},\cdots,C^{*n}$
Other (M - n) Individual Detectors
User Labeling
Mixed Feature Vector with (M - n) machine-predicted probabilities and n ground-truth labels
Context-Based Concept Fusion Learning
Active CBCF Detection Result $P_{ACBCF}(y_I^{*1}=1|I),\cdots,P_{ACBCF}(y_I^{*M}=1|I)$

Figure 3.6: The work flow of Active CBCF.

Different from passive labeling that randomly selects concepts, our Active CBCF actively selects concepts for users' annotation. In the following subsections, we present a statistical criterion to select concepts based on mutual information between concepts, and the error rate and entropy of individual detection results.

3.5.2 Maximizing Mutual Information

The average Bayes classification error for concept detection is $\overline{\mathcal{E}} = \frac{1}{M}\sum_{i=1}^{M}\mathcal{E}(C^i)$, where $\mathcal{E}(C^i)$ is the error rate for concept C^i. The problem is to find an optimal $\mathbf{C}^*$ with a small size n so that the labeled information can minimize $\overline{\mathcal{E}}$ of the rest of concepts. Directly searching for $\mathbf{C}^*$ to minimize $\overline{\mathcal{E}}$ is practically infeasible. Instead, we try to minimize the upper bound of $\overline{\mathcal{E}}$, which can be derived as [77]:

$$\overline{\mathcal{E}} \leq \frac{1}{2M}\sum\nolimits_i H(C^i) - \frac{1}{2M}\sum\nolimits_i I(C^i; C^{*(1:n)}) \tag{3.6}$$

where $C^{*(1:n)} = \{C^{*1}, \dots, C^{*n}\}$. Minimizing the upper bound equals maximizing the mutual information $\sum_i I(C^i; C^{*(1:n)})$. $I(C^i; C^{*(1:n)})$ can be written as [231]:

$$I(C^i; C^{*(1:n)}) = \sum_{k=1}^{n} I(C^i; C^{*k}) - \sum_{k=2}^{n}\left[I(C^{*k}; C^{*(1:k-1)}) - I(C^{*k}; C^{*(1:k-1)}|C^i)\right]$$

The above expression can be further simplified as follows:

$$I(C^{*k}; C^{*(1:k-1)}) - I(C^{*k}; C^{*(1:k-1)}|C^i) = H(C^i) + H(C^{*k})$$

So maximizing $\sum_i I(C^i; C^{*(1:n)})$ equals maximizing:

$$\sum_i \sum_{k=1}^{n} I(C^i; C^{*k}) - \sum_i \sum_{k=2}^{n} H(C^{*k}) \tag{3.7}$$

With the objective function of Eqn. (3.7), we can implement a sequential concept selection process as follows:

1. Select the 1st optimal C^{*1} with largest $g_1^* = \sum_i I(C^i; C^{*1})$
2. Select the k-th optimal C^{*k}, $k > 1$, with largest $g_k^* = \sum_i \Big[I(C^i; C^{*k}) - H(C^{*k}) \Big] = -\sum_i H(C^{*k}|C^i)$

The selection criteria listed above are actually quite intuitive – the best concept has the highest total mutual information with the rest of concepts, and the next optimal concepts are those having low conditional entropy (*i.e.*, high correlations with the rest of concepts).

3.5.3 Further Consideration

When g_k^* is large, accurate labels for concept C^{*k} are desired to better estimate the rest of the concepts. However if C^{*k} can be accurately detected, user's labeling will not make a big difference. In this case, it will be more beneficial to ask user to label a different concept that has low detection accuracy. Thus we need to take into consideration the robustness of the individual detectors. Furthermore, if an individual detector is known to be reliable through validation, we may assume that the estimated label of this detector is close to the true label. Then user's labeling will not make big difference either. Those informative concepts whose individual detectors are not confident about their estimations should get higher priorities for user annotation.

From validation, we get the validation error rate $E_I(C^i)$ of the individual detector for every concept C^i, which estimates the robustness of the individual detector. The confidence of an individual detector about its hypothesis can be measured by the entropy of its output:

$$H_I(C^i|x) = -P(y_I^i = 1|I) \log P(y_I^i = 1|I) - P(y_I^i = 0|I) \log P(y_I^i = 0|I) \tag{3.8}$$

Note that this measure is computed for each image, and thus it is image dependent. Let $\tilde{g}_k^*$ be the normalized version of g_k^* (normalized to [0,1]). To take into consideration the mutual information among concepts as well as the robustness and confidence of individual detectors, our criterion selects "key" concepts as follows: select C^{*k} with the largest $G_k{}^*$, where

$$G_k{}^* = \alpha \tilde{g}_k^* + (1-\alpha) E_I(C^{k^*}) H_I(C^{*k}|I) \tag{3.9}$$

Note the inclusion of the last term makes the selection criterion image-dependent. This is consistent with intuition – we should ask users different questions for different images.

3.6 Experiments: Active CBCF

We use the TRECVID 2005 development set [170] again to evaluate our algorithm. The whole data set is separated into five subsets for training independent detectors, training CBCF models, parameter selections, and final testing (shown in Table 3.3). The pairwise co-occurrences of concepts in the keyframes are counted based on the training set, validation set, selection set 1 and selection set 2, according to which the mutual information is calculated. Again, our individual concept detectors are SVM-based classifiers over simple image features extracted from keyframes of videos. The outputs of the SVM classifiers are transformed into probabilities through a sigmoid function.

Table 3.3: The data sets for experiments to evaluate Active CBCF.

Name	**Size**	**Usage**
Training Set	41837 keyframes	train independent SVMs
Validation Set	4515 keyframes	learn RBF kernel for SVM
Selection Set 1	6021 keyframes	learn CBCF detectors
Selection Set 2	3011 keyframes	learn β in Eqn. (3.9)
Test Set	6506 keyframes	performance evaluation

To show the effectiveness of our Active CBCF method, it is compared with three other methods: the passive labeling (that randomly selects concepts for user to label), the baseline

CBCF without active labeling, and the individual detector without concept fusion. Note that with user's interaction, some keyframes are labeled for each concept. If we calculate precision or recall based on all the test keyframes, the comparison will be unfair to the algorithms without user labeling. To avoid this problem, we compare the algorithms as follows: for each concept C^i, those keyframes selected to be labeled by either our Active CBCF or random labeling are taken out, and the rest keyframes are included for calculating precision or recall.

From the previous experiments (*i.e.*, Table 3.2), we select 26 semantic concepts from the 39 LSCOM-Lite concepts to evaluate our Active CBCF algorithm. Fig. 3.7 gives the individual AP over each of the selected 26 concepts and the overall averaged MAP. The figure shows that out of the selected 26 concepts, 24 have performance improvements using our Active CBCF. Furthermore, the improvements for most concepts are significant, *e.g.*, 20% for "building", 2549% for "government-leader", 1054% for "corporate-leader", 19% for "military", 24% for "mountain", 1026% for "office", 25% for "people-marching", 23% for "road", 17% for "sky", 16% for "urban", and 11% for "waterscape-waterfront", on a relative basis. The MAP of Active CBCF on the entire 26 concepts is 50.9%, which outperforms random labeling by 6.2%, outperforms baseline CBCF by 3.4%, and outperforms the individual detector by 9.2%. Moreover the Active CBCF yields more stable results than both random labeling and CBCF. For example, over some of the selected concepts (*e.g.*, "airplane", "computer-tv-screen", and "crowd") the CBCF performance deteriorates severely. In comparison, the proposed Active CBCF remains effective consistently across all selected concepts.

In addition, we evaluate the performance when different numbers of concepts are labeled by the user for each keyframe. The MAPs on the selected 26 concepts are given in Fig. 3.8, which shows that compared with random labeling, Active CBCF is consistently superior when more and more concepts are labeled by the user. Also, on the test set, when 1 concept is labeled for each keyframe, outdoor is selected to be labeled for 67% of keyframes. When 2 concepts are labeled, besides "outdoor", "airplane" is the second most popular concept selected for 42% of keyframes. When 3 concepts are labeled, besides "outdoor" and "airplane", "court" is selected for 36% of keyframes as the third most popular concept.

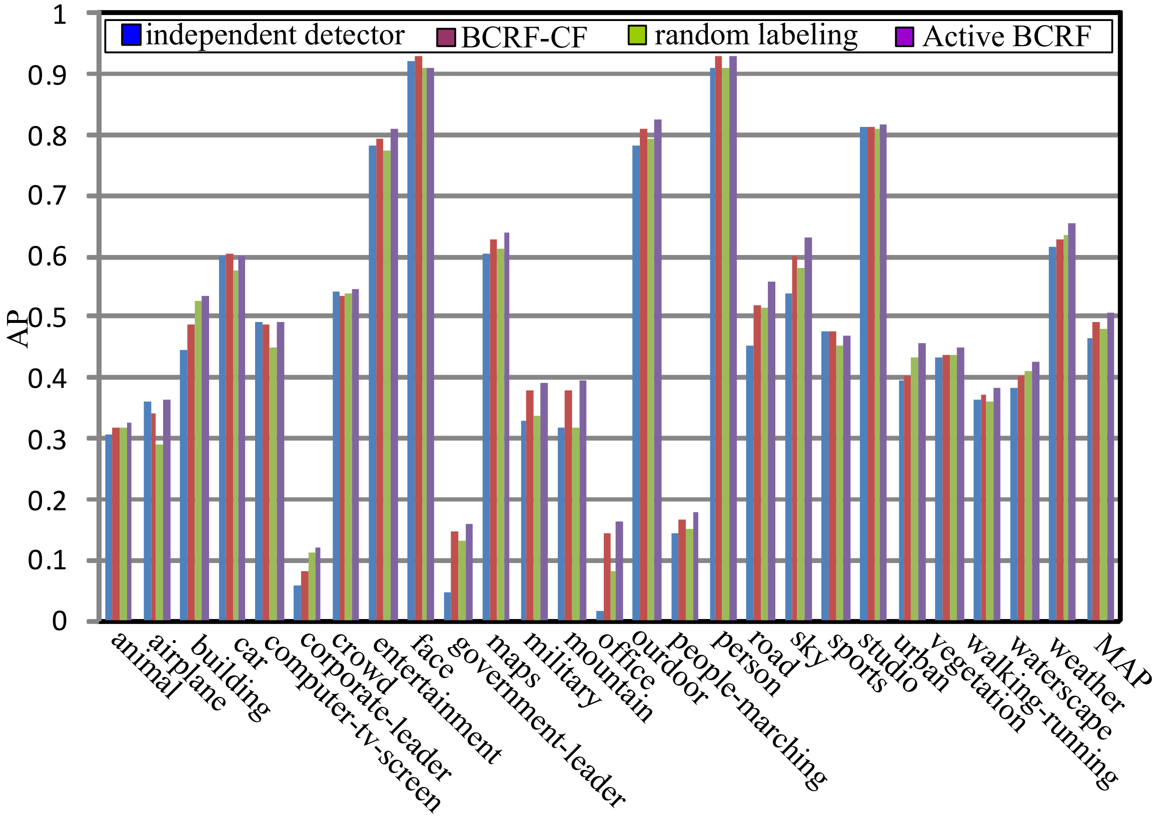


Figure 3.7: Performance with 1 concept labeled for each keyframe.

Fig. 3.9 gives some example keyframes where our algorithm can effectively select the most informative concepts for the user to label. For example, for the "airplane" image, our algorithm selects "outdoor" and "airplane" for the user to label.

3.7 Unique Contributions

In this chapter we propose to model the inter-conceptual relations by a CRF that takes as input detection results from independent detectors and computes updated marginal probabilities as improved detection results. A Boosted CRF framework is developed to optimize the discriminative cost function and avoid the difficulty of designing compatibility potentials. A concept selection criterion is developed to predict which concepts will benefit from CBCF. In addition, we study the interesting "20 questions problem" for semantic concept

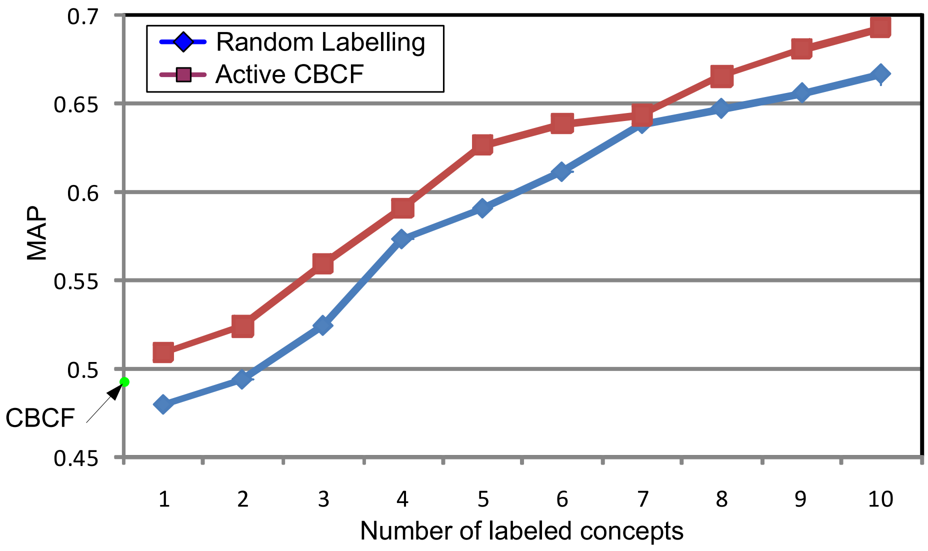


Figure 3.8: MAP with different numbers of labeled concepts.

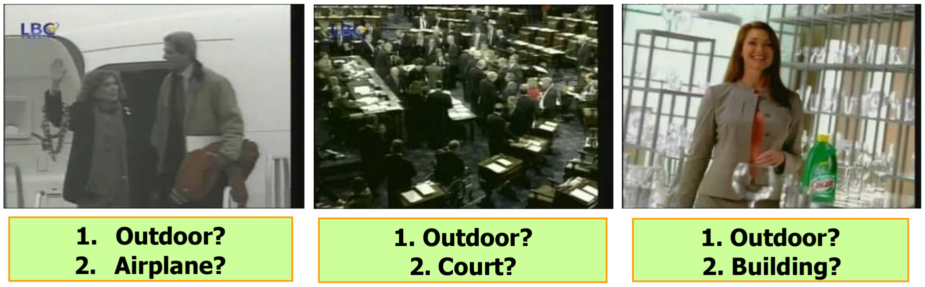


Figure 3.9: Example images where Active CBCF can effectively select the most informative concepts for the user to label.

detection. We incorporate user's interaction to annotate a small number of key concepts per image, which are then used to improve detection of other concepts. We propose a simple yet accurate method to choose the most informative concepts for the user to label. Experiments over the TRECVID 2005 development data set have confirmed the effectiveness of our BCRF-CF method and the concept prediction method, and that the Active CBCF paradigm can further improve the concept detection performance by effectively using users' annotation.

Chapter 4

Multi-Concept: Sharing Kernels and Classifiers

4.1 Introduction

In the previous chapter, we explore multi-concept learning by modeling the inter-conceptual relationships through a CBCF framework. In this chapter we develop a novel multi-class concept detection method based on kernel sharing and joint learning. By sharing "good" kernels among different concepts, accuracy of individual detectors can be improved; by joint learning of common detectors across different classes, required kernels and computational complexity for detecting individual concepts can be reduced.

As discussed in Section 2.2, among various approaches for concept detection, kernel-based discriminative methods, *e.g.*, classifiers based on SVM [230], have been demonstrated effective for detecting generic concepts in many previous works [2, 64, 65, 85, 119, 171]. A large variety of kernels can be constructed based on different descriptors extracted from images, *e.g.*, color descriptor [257], texture descriptor [176], SIFT descriptor [64], or the BoF representation [54, 100, 235, 259]. In BoF, images are represented by a visual vocabulary constructed by clustering the original local descriptors into a set of visual words. Based on these representations, recently, the pyramid matching algorithms [64, 65, 119] have also been proposed to fuse information from multiple resolutions in the spatial domain or feature space. In these approaches, vocabularies constructed from the positive training data of each concept are used to compute kernels for different concepts, and detectors for each individual concept are independently learned in an one vs. all manner.

In this chapter, instead of building independent binary one vs. all concept detectors, we explore the multi-label concept detection problem based on joint learning of shared vocabularies/kernels and shared detectors. Different kernels designed for individual concepts are shared by multiple concept detectors, and common detectors for multiple concepts are jointly learned. The motivation is as follows: concepts are generally related to each other, *e.g.*, the “boat-ship” concept is related to the “waterscape-waterfront” concept. By sharing vocabularies/kernels from interrelated concepts, each individual concept can be enhanced by incorporating the descriptive power from others. For example, as shown in Fig. 4.1, the “boat-ship” concept is very difficult to recognize, due to its diverse visual appearances and thus the lack of strong visual words. Kernels computed using such low-quality vocabularies from “boat-ship” itself usually do not have adequate discriminative power and thus yield unsatisfactory recognition performance. On the other hand, images from “waterscape-waterfront” have good consistency in visual appearances, and high quality vocabularies can be learned to describe the characteristics of waterscape. Kernels calculated using the good vocabularies from “waterscape-waterfront” can be used to help detect the “boat-ship” images which often co-occur with the “waterscape-waterfront” concept. Moreover, common classifiers among different concepts can be built by aggregating limited training data from individual classes. As will be shown in Section 4.3 and 4.4, the complexity in building classifiers can be greatly reduced by such sharing.

To jointly learn kernel-based detectors and to share kernels from different concepts, we propose a joint boosting algorithm to automatically select the optimal kernels and the subsets of sharing concepts in an iterative boosting process. We modify the traditional JointBoost algorithm [223] to incorporate the general Real AdaBoost method [59], so that kernel-based classifiers can be applied for joint learning. Our framework is flexible where any type of kernel can be incorporated for multi-class concept detection.

In addition, we propose a new kernel construction algorithm, namely Vocabulary-Spatial Pyramid Matching (VSPM), which constructs multi-resolution visual vocabularies by hierarchical clustering and computes kernels based on multi-resolution spatial matching. This is essentially an extension of the previous spatial-pyramid matching [119] and vocabulary-guided pyramid matching [65] methods. Although impressive performance has been obtained individually by these two previous approaches, there are some issues as indicated in [119] and our experiments. It is difficult for spatial pyramid matching to automatically choose an appropriate vocabulary resolution for each specific concept. Vocabulary-guided pyramid matching does not consider the spatial layout of local features [119, 174]. In our work, we combine multi-resolution spatial matching and multi-layer vocabulary fusion to generate multiple vocabularies/kernels for each concept.

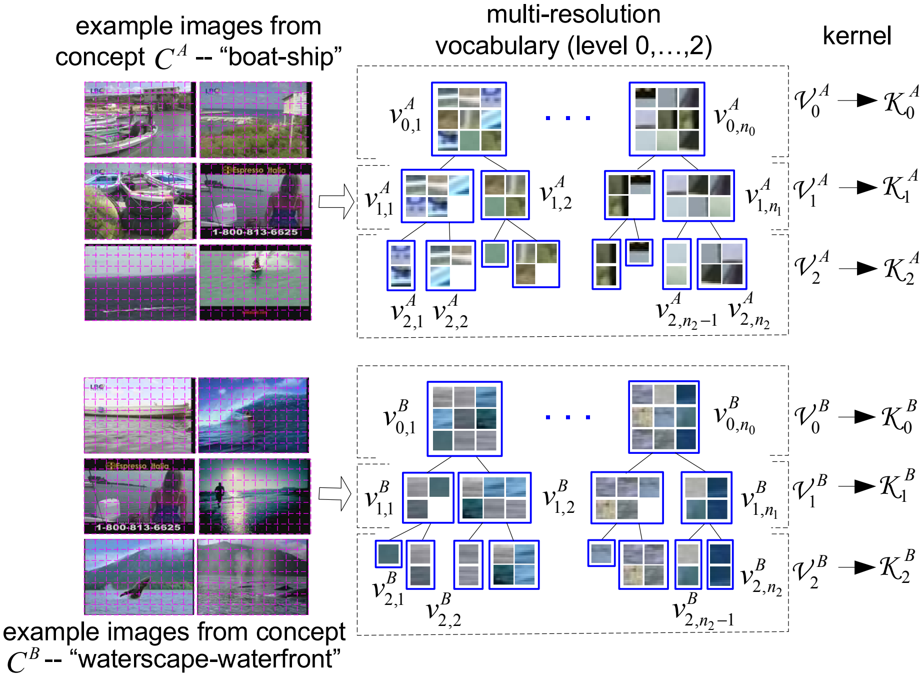

example images from concept C^A -- "boat-ship"
multi-resolution vocabulary (level 0,...,2)
kernel
$v^A_{0,1}$
v^A_{0,n_0}
$v^A_{1,1}$
$v^A_{1,2}$
v^A_{1,n_1}
$v^A_{2,1}$
$v^A_{2,2}$
v^A_{2,n_2-1}
v^A_{2,n_2}
$\mathcal{V}^A_0 \rightarrow \mathcal{K}^A_0$
$\mathcal{V}^A_1 \rightarrow \mathcal{K}^A_1$
$\mathcal{V}^A_2 \rightarrow \mathcal{K}^A_2$
$v^B_{0,1}$
v^B_{0,n_0}
$v^B_{1,1}$
$v^B_{1,2}$
v^B_{1,n_1}
$v^B_{2,1}$
$v^B_{2,2}$
v^B_{2,n_2-1}
v^B_{2,n_2}
$\mathcal{V}^B_0 \rightarrow \mathcal{K}^B_0$
$\mathcal{V}^B_1 \rightarrow \mathcal{K}^B_1$
$\mathcal{V}^B_2 \rightarrow \mathcal{K}^B_2$
example images from concept C^B -- "waterscape-waterfront"

Figure 4.1: Examples of vocabulary/kernel construction. For each concept, *e.g.*, C^A, SIFT features are extracted from uniform grids over each training image from this concept. Hierarchical clustering is applied to construct vocabularies with different resolutions: $\mathcal{V}^A_1, \ldots, \mathcal{V}^A_L$ (L is vocabulary resolution and $L = 2$ in this example), where $\mathcal{V}^A_l$ consists of n_l visual words $\{v^A_{l,1}, \ldots, v^A_{l,n_l}\}$. Then vocabulary-spatial pyramid match kernels $\mathcal{K}^A_0, \ldots, \mathcal{K}^A_L$ are computed (see Section 4.2.2 for details), one kernel for each vocabulary.

We evaluate our method in detecting 12 concepts (objects, scenes, *etc.*) over the TRECVID 2005 data set [170]. Significant performance gains are achieved – 10% in MAP and up to 34% AP for some concepts like “maps”, “building”, and “boat-ship”. Extensive analysis of the results also reveals interesting and important underlying relations among object classes. In addition, we avoid the linear increase of complexity at run time as the number of concepts grows, by sharing detectors among concepts.

4.2 Construction of Vocabulary/Kernel Pool

We use the same terminology as those in Section 2.1: $C^1, \ldots, C^M$ are M semantic concepts; $\mathcal{D} = \{I, \mathbf{y}_I\}$ is the data set, where $\mathbf{y}_I = [y_I^1, \ldots, y_I^M]$ is ground-truth concept labels of image I, with $y_I^i = 1$ or -1 representing the presence or absence of concept C^i in image I; $\mathcal{F}$ is a feature space (*e.g.*, the 128 dimensional SIFT feature space used in [119]). In the following subsection, we will introduce the procedure of constructing multi-resolution vocabularies, followed by the details of our VSPM kernels.

4.2.1 Vocabulary Construction

The BoF representation provides an effective framework for partitioning the original feature space $\mathcal{F}$ to establish a vocabulary of prototypical image features [54, 100, 235, 259]. These prototypes are called visual words. Recently, in [65] multi-resolution vocabularies are constructed by hierarchical clustering. Images are represented with multi-level histograms defined over multiple vocabularies. Kernel matching between images is computed based on histogram intersection. As discussed in [65], vocabularies established from hierarchical clustering take advantage of the underlying structure of the feature space.

In this work, we adopt the hierarchical vocabulary construction procedure and further apply the pyramid matching process in the spatial domain. We incorporate the spatial layout of local descriptors that have been shown to be effective for generic concept detection [119, 174]. As shown in Fig. 4.1, for each concept C^i, the local descriptors from the positive training samples are aggregated together, and hierarchical clustering is applied to generate $L+1$ vocabularies that partition the feature space $\mathcal{F}$ into a pyramid of non-uniformly shaped regions. Let $\mathcal{V}_l^i = \{v_{l,1}^i, \ldots, v_{l,n_l}^i\}$ denote the vocabulary at level l, $0 \leq l \leq L$, where $v_{l,g}^i$ represents the g-th visual word in vocabulary $\mathcal{V}_l^i$. Based on these multi-level vocabularies, in the next subsection, a VSPM method will be developed to construct multi-resolution kernels for each individual concept.

4.2.2 Kernel Construction

For each concept C^i, with each vocabulary $\mathcal{V}_l^i$, we construct a VSPM kernel based on spatial matching. We adopt the spatial pyramid matching algorithm proposed in [119], which computes correspondences between images by repeatedly subdividing images into spatial pyramids and computing histogram intersections at multiple spatial resolutions. This method achieves robust performance for generic scene detection by incorporating the spatial layout information of local descriptors [119]. In this work, we extend the original spatial pyramid matching to incorporate multi-resolution vocabularies. As both evidenced by experiments in [119] and our experiments in Section 4.4 (*i.e.*, Table 4.1 and Fig. 4.5), different concepts favor different resolutions of vocabularies, and it is generally difficult to determine one optimal resolution for detecting all concepts. In our proposed method, we construct a pool of vocabularies over multiple resolutions and develop an automatic mechanism for selecting the optimal ones.

Specifically, given a concept C^i and a vocabulary $\mathcal{V}_l^i$, the spatial pyramid matching method [119] hierarchically quantizes the spatial coordinates of images to $S+1$ different scales, with 4^s discrete blocks at each scale s, $0 \leq s \leq S$. Then for each image I, an n_l-dimension histogram $H_{s,l,k}^i(I) = [h_{s,l,k,1}^i(I), \dots, h_{s,l,k,n_l}^i(I)]$ is calculated from block k at spatial scale s, using a certain vocabulary $\mathcal{V}_l^i$, where $h_{s,l,k,g}^i(I)$ is the element bin corresponding to visual word $v_{l,g}^i$. The number of matches at spatial scale s between two images I_p and I_q is computed by histogram intersection:

$$\mathcal{I}_{s,l}^i(I_p, I_q) = \sum_{k=1}^{4^s} \sum_{g=1}^{n_l} \min\left\{h_{s,l,k,g}^i(I_p), h_{s,l,k,g}^i(I_q)\right\}$$

Since the matches found at scale s include all the matches at scale $s+1$, when we go down from fine (scale $s+1$) to coarse (scale s), the new matches found is given by $\mathcal{I}_s - \mathcal{I}_{s+1}$, $0 \leq s \leq S-1$. Assume that each scale s is associated with a weight $\frac{1}{2^{S-s}}$, which favors matches found in finer cells because they involve increasingly similar features. The final

VSPM kernel defined by vocabulary $\mathcal{V}_l^i$ is given by:

$$\begin{aligned}\mathcal{K}_l^i(I_p, I_q) &= \mathcal{I}_{S,l}^i(I_p, I_q) + \sum_{s=0}^{S-1} \frac{\mathcal{I}_{s,l}^i(I_p, I_q) - \mathcal{I}_{s+1,l}^i(I_p, I_q)}{2^{S-s}} \\ &= \frac{1}{2^S}\mathcal{I}_{0,l}^i(I_p, I_q) + \sum_{s=1}^{S} \frac{1}{2^{S-s+1}}\mathcal{I}_{s,l}^i(I_p, I_q) \end{aligned} \tag{4.1}$$

For each concept C^i, we have $L+1$ VSPM kernels $\mathcal{K}_0^i, \ldots, \mathcal{K}_L^i$ defined by $L+1$ vocabularies. Instead of training one vs. all detectors independently, we propose to share various kernels from different concepts across multiple detectors. Specifically, all VSPM kernels from different concepts are aggregated together to generate a kernel pool $\mathbf{K}$. In the next section, a joint boosting algorithm is developed, which provides a systematic data-driven framework to select optimal kernels from $\mathbf{K}$ to be shared and the optimal sets of concepts that share them.

Note that it is also possible to use these VSPM kernels in the one vs. all manner similar to traditional concept detection approaches [65, 119] (as shown in Fig. 4.2 (a)). One method is to combine the $L+1$ kernels $\mathcal{K}_0^i, \ldots, \mathcal{K}_L^i$ from each C^i into one ensemble kernel:

$$\mathcal{K}^i(I_p, I_q) = \sum_{l=0}^{L} \omega_l \mathcal{K}_l^i(I_p, I_q) \tag{4.2}$$

Weight ω_l can be heuristically defined as in [65]. Then this ensemble kernel can be used to train one vs. all classifiers to detect concept C^i independently from other concepts. Another way is to directly apply traditional boosting algorithms (*e.g.*, Real AdaBoost [59]) to train one vs. all detectors for each individual concept C^i by selecting appropriate kernels from $\mathcal{K}_0^i, \ldots, \mathcal{K}_L^i$ through boosting iterations. In our experiments in Section 4.4, we will use these two approaches as baselines to compare with our joint boosting algorithm.

4.3 Joint Learning with Joint Boosting

The idea of joint boosting is to share features and classifiers over multiple categories. Most of the existing concept detection approaches [54, 64, 65, 119, 176] treat each individual concept C^i independently from other concepts. As shown in Fig. 4.2 (a), these one vs. all detectors train a binary classifier to detect each concept C^i, based on VSPM kernels

$\mathcal{K}_0^i, \ldots, \mathcal{K}_L^i$ derived from training data associated with C^i only. While in joint boosting (shown in Fig. 4.2 (b)), multi-class concepts are detected simultaneously by sharing a kernel pool consisting of all kernels from various concepts and sharing classifiers during boosting iterations. For example, in Fig. 4.2 (b) we first use a kernel $\mathcal{K}^*(1)$ (an optimal kernel chosen from $\mathbf{K}$) to learn a binary classifier to separate C^A, C^B from the background, *i.e.*, C^A and C^B share this classifier and kernel. Then using $\mathcal{K}^*(2)$, a binary classifier further picks out C^A. In the next subsections, we will describe the boosting framework followed by our joint boosting method.

4.3.1 Learning with Boosting

The multi-class concept detection problem can be formulated as minimizing a multi-class cost function:

$$\mathcal{L} = \sum_{i=1}^{M} \sum_{(I,\mathbf{y}_I)\in\mathcal{D}} w_I^i e^{-y_I^i Q_I^i} \tag{4.3}$$

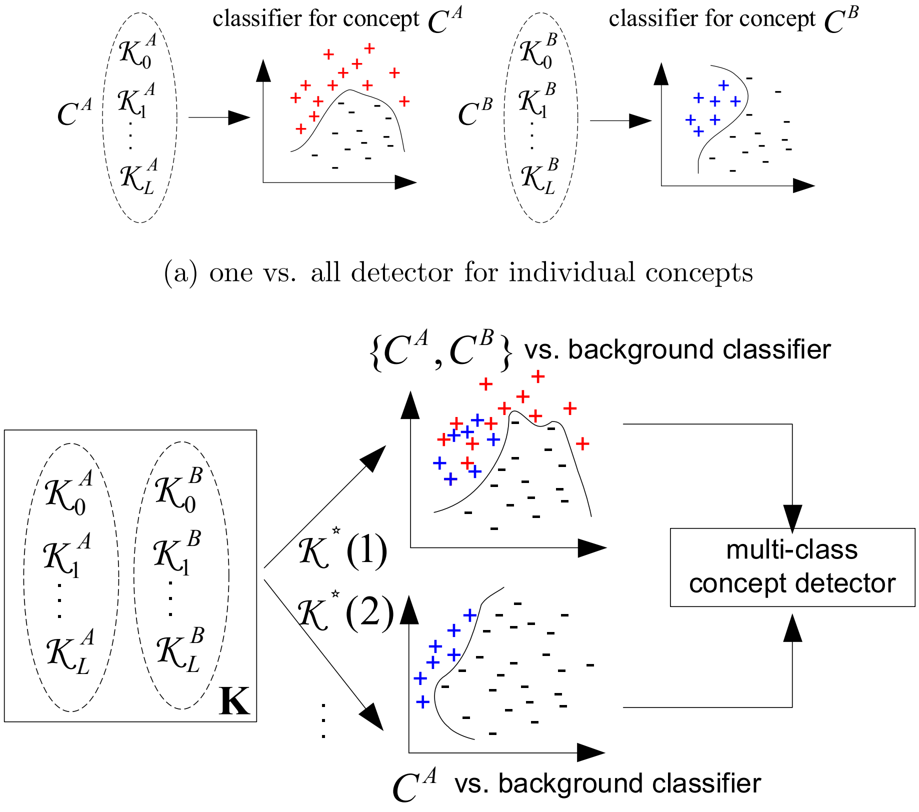
where y_I^i is the ground-truth label of concept C^i for image I, and w_I^i is the sample weight of image I for concept C^i. Q_I^j is a discriminative function (the classification result) we want to learn. In one vs. all independent detection methods, the multi-class concept detection problem is converted into a set of binary classification problems. That is, $\mathcal{L}$ is approximately optimized by minimizing the following $\mathcal{L}^i$ for each concept C^i independently:

$$\mathcal{L}^i = \sum_{(I,\mathbf{y}_I)\in\mathcal{D}} w_I^i e^{-y_I^i Q_I^i} \tag{4.4}$$

In contrast, in the joint learning scenario, we optimize $\mathcal{L}$ jointly by training a multi-class classifier. Since concepts are usually related to each other, joint learning exploits such correlations and can better optimize $\mathcal{L}$ than independent training approaches.

The well-known boosting framework effectively combines many weak classifiers to form a powerful ensemble classifier. In the training stage, training samples are re-weighted according to the training error from previous weak learners, and classifiers trained later are forced to focus on harder samples with higher weights. The boosting procedure is formulated by assuming an additive model for Q_I^i:

$$Q_I^i(T) = \sum_t q_I^i(t) \tag{4.5}$$


classifier for concept C^A
C^A
$\mathcal{K}_0^A$
$\mathcal{K}_1^A$
$\mathcal{K}_L^A$
classifier for concept C^B
C^B
$\mathcal{K}_0^B$
$\mathcal{K}_1^B$
$\mathcal{K}_L^B$
$\{C^A, C^B\}$ vs. background classifier
$\mathcal{K}^*(1)$
$\mathcal{K}^*(2)$
K
multi-class concept detector
C^A vs. background classifier

(a) one vs. all detector for individual concepts

(b) multi-class joint detector for kernel and classifier sharing

Figure 4.2: Illustration of learning procedures. (a) one vs. all detectors treat each individual concept independently, *e.g.*, C^A is classified only based on kernels $\mathcal{K}_0^A, \ldots, \mathcal{K}_L^A$ derived from training data associated with this concept, independently from other concepts. (b) in joint boosting, a kernel pool **K** (including kernels from all concepts) is shared by different detectors.

where each $q_I^i(t)$ is the hypothesis generated by the t-th weak learner in iteration t. Several variants of boosting algorithm have been developed, such as AdaBoost [202], Real AdaBoost [59], GentleBoost [59], LogitBoost [59], and boosting feature selection [100, 219]

The original boosting algorithm is designed for binary classification, and several multi-class learning algorithms, *e.g.*, AdaBoost.MO, AdaBoost.MH and AdaBoost.MR [202], have been proposed as extensions. In [223] a JointBoost algorithm is proposed based on GentleBoost. This method tries to find the common feature axes and the corresponding weak learners that can be shared across multiple classes. In [177] the JointBoost algorithm is extended into a Joint-AdaBoost approach that enables incremental additions of new classes.

4.3.2 Joint Boosting and Kernel Sharing

Motivated by the work of [177, 223], in this work we develop our joint boosting algorithm based on Real AdaBoost due to its flexibility for incorporating the powerful kernel-based classifiers for concept detection and its compatibility for our proposed strategy to share kernels in the iterative boosting steps. Specifically, during each iteration t, our joint boosting method selects the optimal subset of concepts $\mathcal{C}^*(t)$ and the corresponding optimal kernel $\mathcal{K}^*(t)$. Positive training samples from any concept in $\mathcal{C}^*(t)$ are treated as positive, and a binary classifier is learned based on $\mathcal{K}^*(t)$ to separate these positive data from the background negative data (samples not belonging to any concept in subset $\mathcal{C}^*(t)$). The optimal $\mathcal{C}^*(t)$ and $\mathcal{K}^*(t)$ are chosen by sequentially searching from every possible subset $\mathcal{C}(t)$ of concepts using every possible kernel $\mathcal{K}(t)$ in the kernel pool $\mathbf{K}$ so that the corresponding classifier has the smallest error on the weighted training set for all concepts.

In the Real AdaBoost formulation [59], during each iteration t, we want to minimize the cost function $\mathcal{L}$ in Eqn. (4.3) with respect to $q_I^i(t)$, $\mathcal{C}(t)$ and $\mathcal{K}(t)$ as follows:

$$\mathcal{L}(q_I^i(t), \mathcal{C}(t), \mathcal{K}(t)) = \sum_{i=1}^{M} \sum_{(I,\mathbf{y}_I)\in\mathcal{D}} w_I^i(t) e^{-y_I^i\left(Q_I^i(t-1)+q_I^i(t)\right)} \tag{4.6}$$

where $q_I^i(t)$ has the following form:

$$q_I^i(t) = \begin{cases} \frac{1}{2}\log\frac{p^t(y_I^i=1|I)}{p^t(y_I^i=-1|I)} & , \ C^i \in \mathcal{C}(t) \\ k_c^i(t) & , \ others \end{cases} \tag{4.7}$$

$p^t(y_I^i = 1|I)$ is the posterior probability of the positive detection of concept C^i generated by the binary weak classifier that separates the positive training samples from the subset $\mathcal{C}(t)$ and the background samples. All concepts $C^i \in \mathcal{C}(t)$ share this binary classifier and have the same $p^t(y_I^i = 1|I)$ value. $k_c^i(t)$ is a constant for concept C^i that is not selected to share the weak learner in iteration t. A natural choice of $k_c^i(t)$ is to use the prior, namely:

$$p^t(y_I^i = 1|I) = \frac{\sum_{(I,\mathbf{y}_i)\in\mathcal{D}} w_I^i(t)\delta(y_I^i = 1)}{\sum_{(I,\mathbf{y}_I)\in\mathcal{D}} w_I^i(t)} \tag{4.8}$$

where $\delta(\cdot)$ is the indicator function. Then $k_c^i(t)$ is given by:

$$k_c^j(t) = \frac{1}{2}\log\frac{\sum_{(I,\mathbf{y}_I)\in\mathcal{D}} w_I^i(t)\delta(y_I^i=1)}{\sum_{(I,\mathbf{y}_I)\in\mathcal{D}} w_I^i(t)\delta(y_I^i=-1)} \tag{4.9}$$

$k_c^j(t)$ depends on the empirical counts of the positive and negative training samples of C^i only, rather than the kernel-based classifier shared with other concepts.

Any type of classifier can be used to generate the posterior probability $p^t(y_I^i=1|I)$ for concepts in $\mathcal{C}(t)$. In this work, we use the SVM classifier for concept detection. To conduct sample re-weighting for the SVM classifier, we use the random sampling method. That is, during each iteration t, for each concept C^i, we form a new training set $\tilde{\mathcal{D}}^i(t)$ generated by randomly sampling training data from the original training set $\mathcal{D}$ with a frequency according to the sample weights $w_I^i(t)$.

The detailed joint boosting algorithm is summarized in Fig. 4.3. Originally, we need to search all possible $N(2^M-1)$ combinations of concepts and kernels, where M is the number of concepts, and N is the number of kernels in $\mathbf{K}$. To reduce the computational complexity, during each iteration t, we adopt the forward selection procedure from [223]: in the first step, we try every concept using every possible kernel in the kernel pool, and then select the best pair of kernel $\mathcal{K}^*(t)$ and concept $C_1^*(t)$ that has the smallest cost based on Eqn. (4.6). Then in the next step, we set $\mathcal{C}_1(t)=\{C_1^*(t)\}$, and for $r=2,\dots,M$, a concept $C_r^*(t)$ is added to $\mathcal{C}_{r-1}(t)$ to form $\mathcal{C}_r(t)=\mathcal{C}_{r-1}(t)\cup C_r^*(t)$ so that the joint detector of subset $\mathcal{C}_r(t)$ has the smallest cost among all possible $C_r(t)$. Finally $\mathcal{C}_M(t)$ will include all concepts. Then from the M candidate subsets $\mathcal{C}_1(t),\dots,\mathcal{C}_M(t)$, we select $\mathcal{C}^*(t)$ with the smallest cost. Note that since our weak learners generate hypotheses according to Eqn. (4.7) and samples are re-weighted in the same way as the original Real AdaBoost: $w_I^i(t)=w_I^i(t-1)e^{-y_I^i q_I^i(t)}$, our joint boosting algorithm inherits the convergence property of the Real AdaBoost algorithm [59]. In other words, the multi-class cost function is guaranteed to improve in every iteration.

The major difference between our joint boosting algorithm and the original JointBoost [223] is that we use Real AdaBoost for sharing kernels, while the original JointBoost uses GentleBoost [59] for sharing feature axes. GentleBoost is a "gentle" version of Real AdaBoost, where the "gentle" Newton stepping is used to improve the cost function $\mathcal{L}$ in each step. In contrast, for Real AdaBoost exact optimization of the cost function is performed in each step. GentleBoost uses logistic regression as the weak learner, which has simpler structure but less flexibility (in terms of weak learner) than Real AdaBoost where any type of classifier can be adopted to generate the hypotheses. To incorporate the powerful kernel-based discriminative classifiers for detecting generic concepts, we build our joint boosting algorithm based on the general Real AdaBoost method.

4.4 Experiments

We use the TRECVID 2005 data set [170] to evaluate our algorithm. Similar to [119], SIFT descriptors are extracted from each 16x16 block over a grid with spacing of 8 pixels. For each concept, the SIFT features from the positive training samples are aggregated and k-means clustering is applied to get a visual vocabulary $\mathcal{V}_0$ at level $l = 0$, consisting of 100 visual words ($n_0 = 100$). Then hierarchical clustering is applied to generate the finer level vocabularies as follows: for each word v_l^i, $0 \leq l \leq L-1$, we perform k-means clustering to divide this word to two sub visual words $v_{l+1,2i}^i$ and $v_{l+1,2i+1}^i$ [1]. That is, the sizes of vocabularies have $n_l = 2n_{l-1}$. In our implementation we have three levels in total, *i.e.*, $L = 2$. So the maximum size of vocabulary is $n_2 = 400$.

From the 39 LSCOM-Lite concepts [164], we use 12 concepts to evaluate our algorithm, including objects (*e.g.*, “flag-us” and “boat-ship”), locations (*e.g.*,“urban”), and scenes (*e.g.*, “waterscape-waterfront” and “vegetation”). One concept, “government-leader”, is also included as a semantically outlier. Although recognition of such highly specific semantic concept seems very difficult, it is made more feasible by the consistent visual scenes (like press conference rooms) often seen in images including government leaders. The concepts were not chosen to specifically optimize the performance gains by the proposed method. Instead, they are chosen to cover diverse semantic concepts and reasonable detection accu-

[1] This is a top-down hierarchical clustering algorithm by k-means. Other hierarchical clustering algorithms can also be used to generate visual vocabularies, such as the bottom-up algorithm to gradually combine similar data into clusters. One advantage of this top-down hierarchical clustering is the easy control of the vocabulary size.

```
algorithm: Joint Boosting
Input: The data set $\mathcal{D}=\{I,\mathbf{y}_I\}$, and the kernel pool: $\mathbf{K}=\left\{\mathcal{K}_0^1,\ldots,\mathcal{K}_L^1,\ldots,\mathcal{K}_0^M,\ldots,\mathcal{K}_L^M\right\}$
($M$ concepts, and $L$ multi-resolution kernels for each individual concept).

• Initialization: $\forall i$, set $w_I^i(0)=1, Q_I^i(0)=0$.
• For iteration $t=1,\ldots,T$
    – get training set $\tilde{\mathcal{D}}^i(t)$ for each concept $C^i$ by randomly sampling $\mathcal{D}$ with a frequency according to weights $w_I^i(t)$.
    – select the optimal kernel $\mathcal{K}^*(t)$ and subset of concepts $\mathcal{C}^*(t)$ in a greedy forward selection procedure:
        ∗ find the optimal pair of kernel $\mathcal{K}^*(t)$ and concept $C_1^*(t)$ to minimize the cost in Eqn. (4.6) by searching from all possible pairs.
        ∗ set $\mathcal{C}_1(t)=\{C_1^*(t)\}$
        ∗ Repeat for $r=2,\ldots,M$
          Get $\mathcal{C}_r(t)=\mathcal{C}_{r-1}(t)\cup C_r^*(t)$ by finding the optimal $C_r^*(t)$ from all possible $C_r(t)\notin\mathcal{C}_{r-1}(t)$ to minimize the cost in Eqn. (4.6).
        ∗ Select the optimal $\mathcal{C}^*(t)$ from all candidate subsets $\mathcal{C}_1(t),\ldots,\mathcal{C}_M(t)$.
    – Train an SVM classifier for $\mathcal{C}^*(t)$ and compute $q_I^i(t)$.
    – For each concept $C^i$:
        ∗ re-weight samples: $w_I^i(t)=w_I^i(t-1)e^{\{-y_I^i q_I^i(t)\}}$.
        ∗ Update decision: $Q_I^i(t)=Q_I^i(t-1)+q_I^i(t)$.
```

Figure 4.3: The joint boosting algorithm based on Real AdaBoost.

racy. The selected 12 concepts have moderate performances (not too high or too low) [2, 170], which makes them suitable for comparing various concept detection algorithms described in this chapter. Example keyframes for the 12 concepts are shown in Fig. 4.4. The training set consists of at most 2000 positive keyframes for each concept and 1000 "background" keyframes (not belonging to any of the 12 concepts). All keyframes are randomly sampled from 90 training videos. 16 videos from the remaining set are randomly selected as the test data. In total, there are 11952 keyframes for training and 6507 keyframes for evaluation.

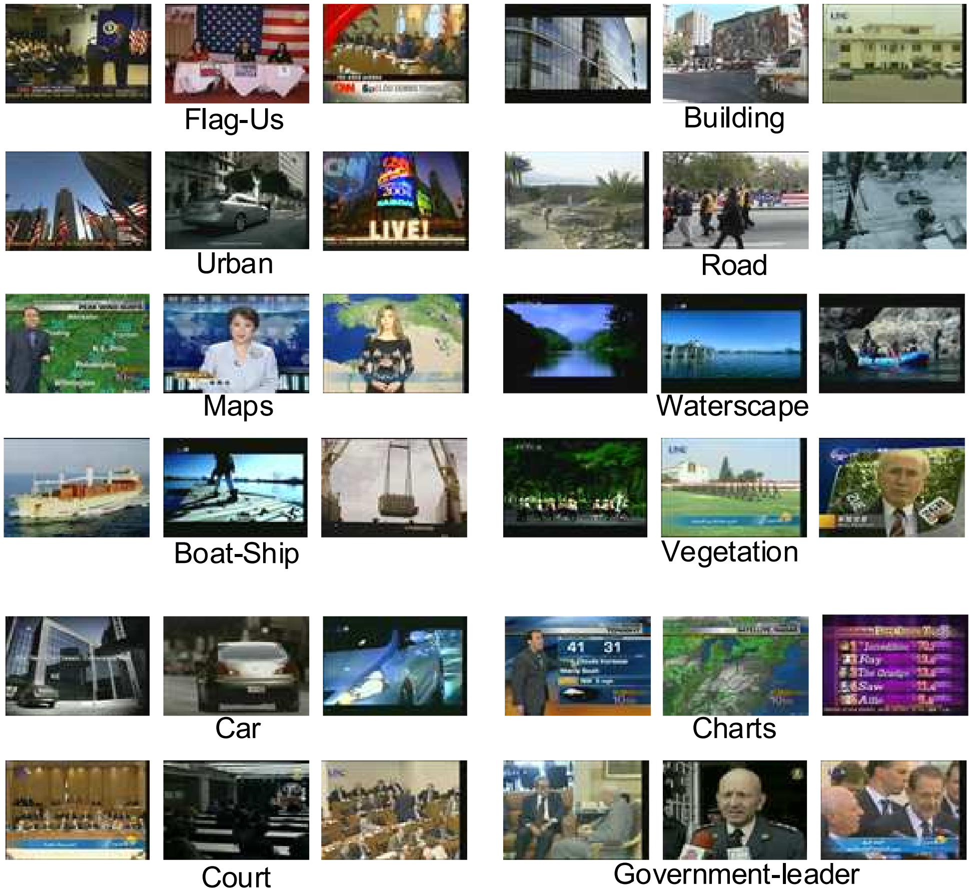


Figure 4.4: The list of 12 concepts and associated example keyframes for evaluating the proposed kernel/classifier sharing method.

4.4.1 Comparison between Different Kernels

First, we evaluate our VSPM kernel using a baseline approach (called VSPM baseline in the following experiments), where for each concept C^i, an ensemble kernel is generated based on Eqn. (4.2) and an one vs. all SVM classifier is trained using the ensemble kernel to detect this concept (as discussed in Section 4.2.2). We compare this baseline with a state-of-the-art approach [119]: SVM classifiers based on the spatial pyramid match kernel using a single vocabulary. Table 4.1 shows the MAP comparison, and Fig. 4.5 shows the per-concept AP performance. From the results, the VSPM baseline slightly outperforms single-vocabulary kernels in terms of MAP, and performs best for 5 concepts in terms of AP, while kernels with single-vocabulary wins over the remaining 7 concepts. Also, no single vocabulary setting outperforms others over all concepts. The single-vocabulary kernel at different levels, $l = 0, 1, 2$, achieves the best performance for 1, 2, and 4 concepts respectively. These results show that different concepts favor different resolutions of kernels, and the VSPM baseline that averages multi-resolution kernels with predefined weights result in slight performance gains.

Table 4.1: MAP: VSPM baseline vs. single-vocabulary kernels (with varying numbers of levels used in the spatial pyramid).

single-vocabulary kernel			**VSPM baseline**
$l=0$	$l=1$	$l=2$	
0.213	0.219	0.242	0.248

4.4.2 Vocabulary Sharing with Joint Boosting

In this experiment, we compare our joint boosting with two baseline approaches: the VSPM baseline presented above, and the regular boosting approach where Real AdaBoost is directly applied to detect individual concepts independently (as discussed in Section 4.2.2). Fig. 4.6 shows the AP performance over each of the 12 concepts. The regular boosting uses 120 weak classifiers (10 iterations for each concept) and joint boosting uses 100 weak classifiers (100 iterations for joint boosting). From the figure we can see that regular boosting actually hurts the performance over most of the concepts. This phenomenon indicates that severe overfitting occurs by using regular boosting due to the small positive training set for many concepts, *e.g.*, only 254 positive samples out of 41847 training samples for “boat-ship”. As demonstrated in [41], with such few and noisy positive training samples, regular

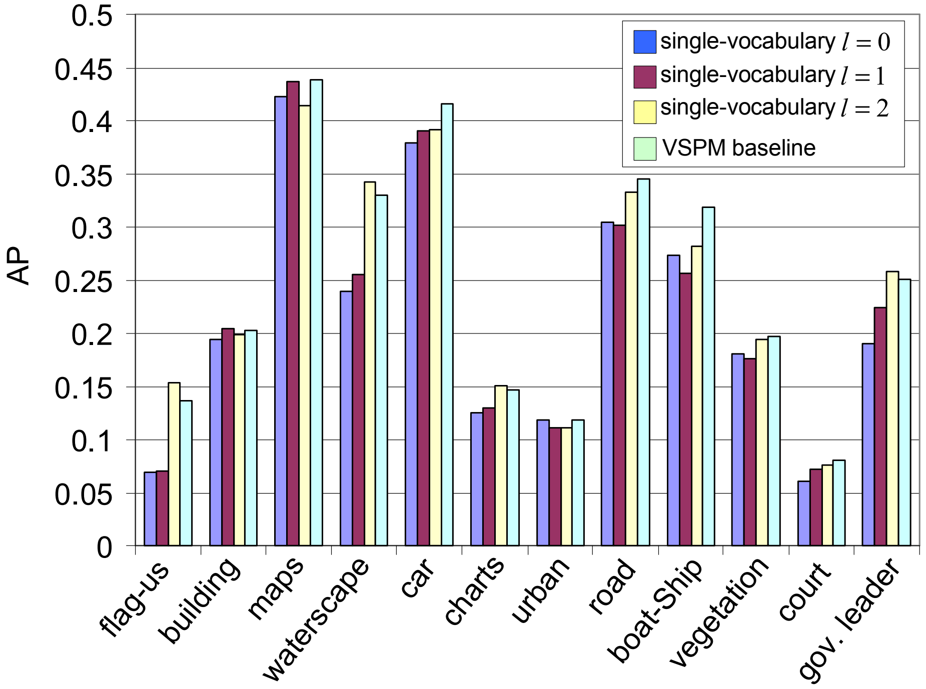


Figure 4.5: Per-concept AP: VSPM baseline vs. single-vocabulary kernels. l indicates the vocabulary resolution level.

boosting usually overfits greatly. Compared with the baselines, our joint boosting method consistently achieves superior performance for most of the concepts even with the limited training data. This phenomenon is consistent with previous literatures. As shown in [177, 223], when the number of training samples are reduced, joint boosting can still get good performance, since through sharing weak learners and kernels among different concepts the required training kernels and samples are reduced for detecting each individual concept [223]. On average, the overall MAP improvement of joint boosting is 24% and 10%, respectively, compared with regular boosting and VSPM baseline. The performance improvements over many concepts are significant, *e.g.*, compared with VSPM baseline, joint boosting achieves the following AP gains: "building" by 30%, "maps" by 34%, "urban" by 10%, "boat-ship" by 17%, "vegetation" by 19%, and "government-leader" by 11%.

In addition, Fig. 4.7 shows the evolution of the MAP performance during boosting iterations. When more weak classifiers are added, joint boosting can achieve better performance, confirming the convergence property of the joint boost algorithm. Note in the current ex-

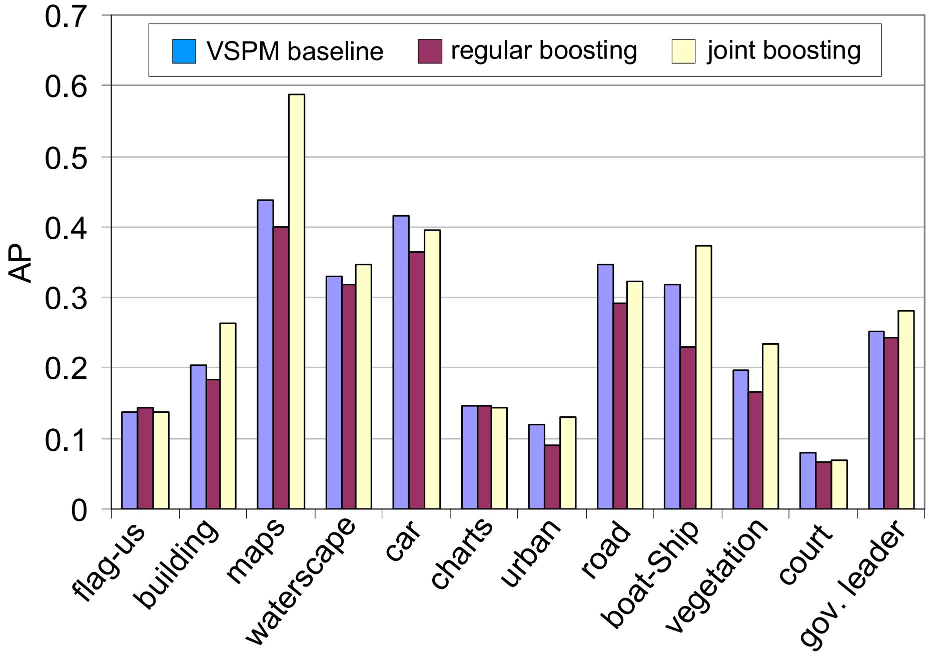


Figure 4.6: Per-concept AP of joint boosting and two baseline approaches: VSPM baseline and regular boosting.

periment, we stop at 100 weak learners, and the performance curve is still rising. This is in contrast to the regular boosting algorithm, where the performance keeps decreasing throughout the iterations.

4.4.3 Exploring Concept Relations

In this subsection, we explore the relationships between the 12 concepts in TRECVID data through analysis of the experimental results from joint boosting. There are 36 kernels in the kernel pool, 3 kernels for each concept (due to 3 levels of k-means clustering used in codebook construction). Only 19 kernels are selected to be used by the joint boosting algorithm. Fig. 4.8 shows the frequency of the usage for each kernel. From the figure, kernels of some concepts, *e.g.*, "maps" and "boat-ship", are seldom used. These concepts are either rare concepts (*i.e.*, very few positive samples), or the supporting keyframes have large diversity in visual appearances (*e.g.*, "boat-ship" as shown in Fig. 4.1). Thus the quality of visual vocabularies from these concepts are poor. This result is consistent with

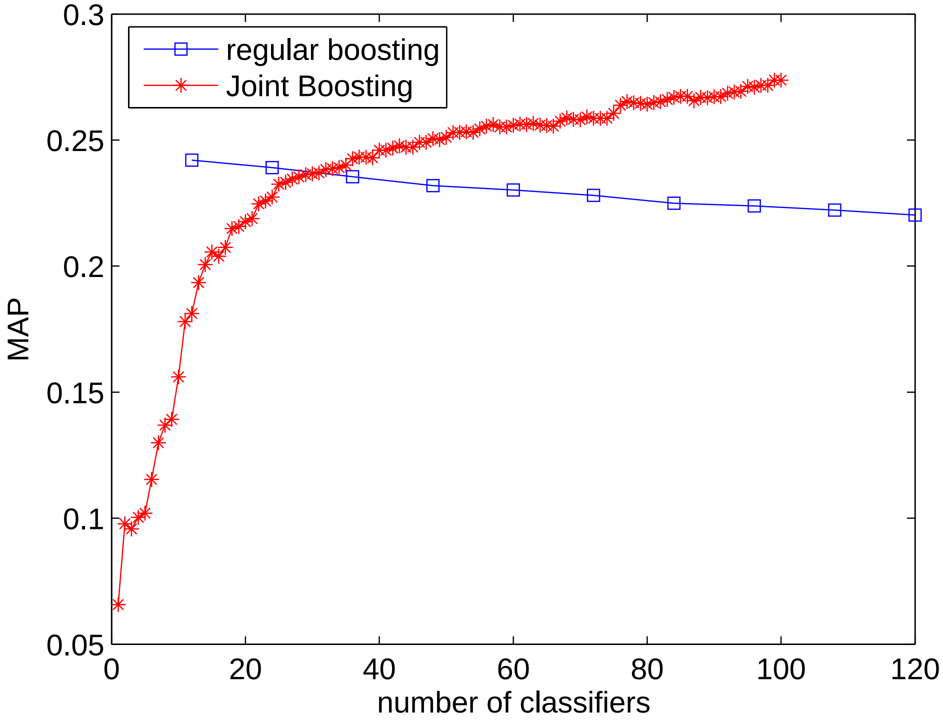


Figure 4.7: Performance evolution of joint boosting and regular boosting with different numbers of weak classifiers.

above experiments that these concepts can benefit a lot using kernels from other concepts, and joint boosting has significant performance improvement for “maps” and “boat-ship” compared with independent detectors.

Fig. 4.9 gives the details about the usage of kernels by each concept. The figure clearly shows the sharing of vocabularies across different concepts. For example, “boat-ship” uses the kernels from “waterscape-waterfront” frequently in addition to its own kernels, and “maps” uses the kernels from many other concepts such as “charts”, “flag-us”, *etc.* This is because of the high correlation between “boat-ship” and “waterscape-waterfront”, and between “maps” and “charts” *etc.* Since keyframes from “waterscape-waterfront” or “charts” *etc.* have high consistency, the high-quality visual words extracted for ‘waterscape-waterfront” or “charts” *etc.* provide important semantic cues for detecting “boat-ship” or “maps” respectively. In addition, Fig. 4.10 shows the AP evolution of the “boat-ship” and

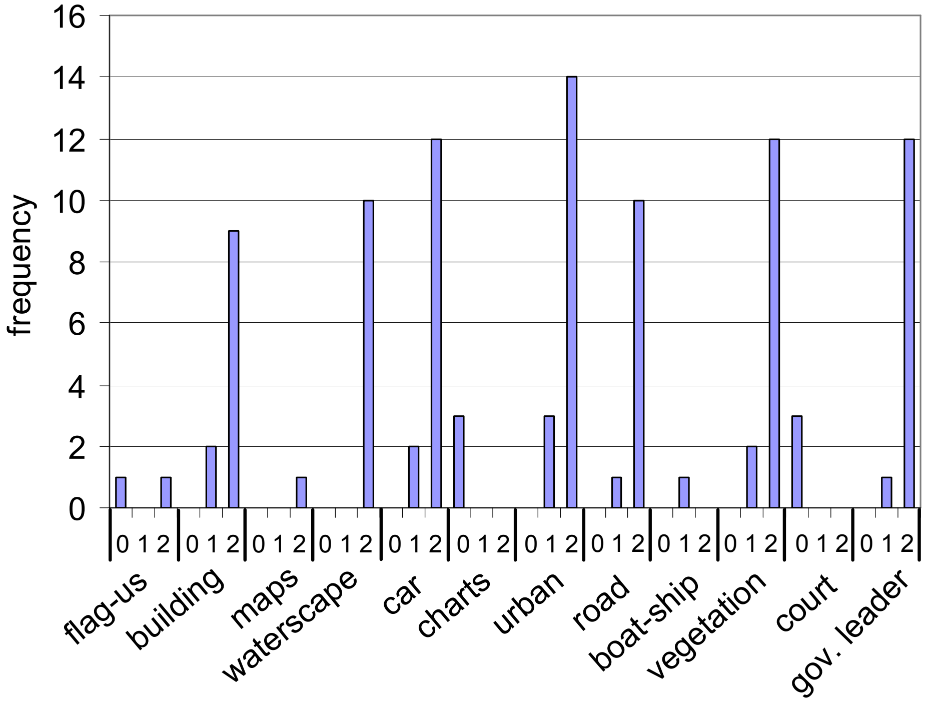


Figure 4.8: Usage of individual kernels by 100 weak classifiers. The horizontal axis shows kernels from different concepts, where numbers 0,1,2 represent multiple resolutions, *i.e.*, l. Kernels from "maps" and "boat-ship" are seldom used. Interestingly, these two concepts are enhanced significantly in joint boosting by leveraging kernels from other related concepts.

"maps" during 100 iterations, where the kernels corresponding to big performance changes are also listed. Note the same kernel may be used multiple times throughout the iterative process. Fig. 4.10 further confirms that the performance of "boat-ship" and "maps" are helped significantly by sharing kernels from "waterscape-waterfront" and "charts" *etc.*, respectively.

4.4.4 Computational Complexity

The test stage of joint boosting is very fast, while its training stage needs more computational time than regular boosting or VSPM baseline without boosting. However, training can be done offline. During the test stage, joint boosting only needs T SVM classification, one for each iteration, to classify M concepts simultaneously. While regular boosting needs $M \cdot T$ SVM classification. In the training stage, for each iteration, joint boosting needs to

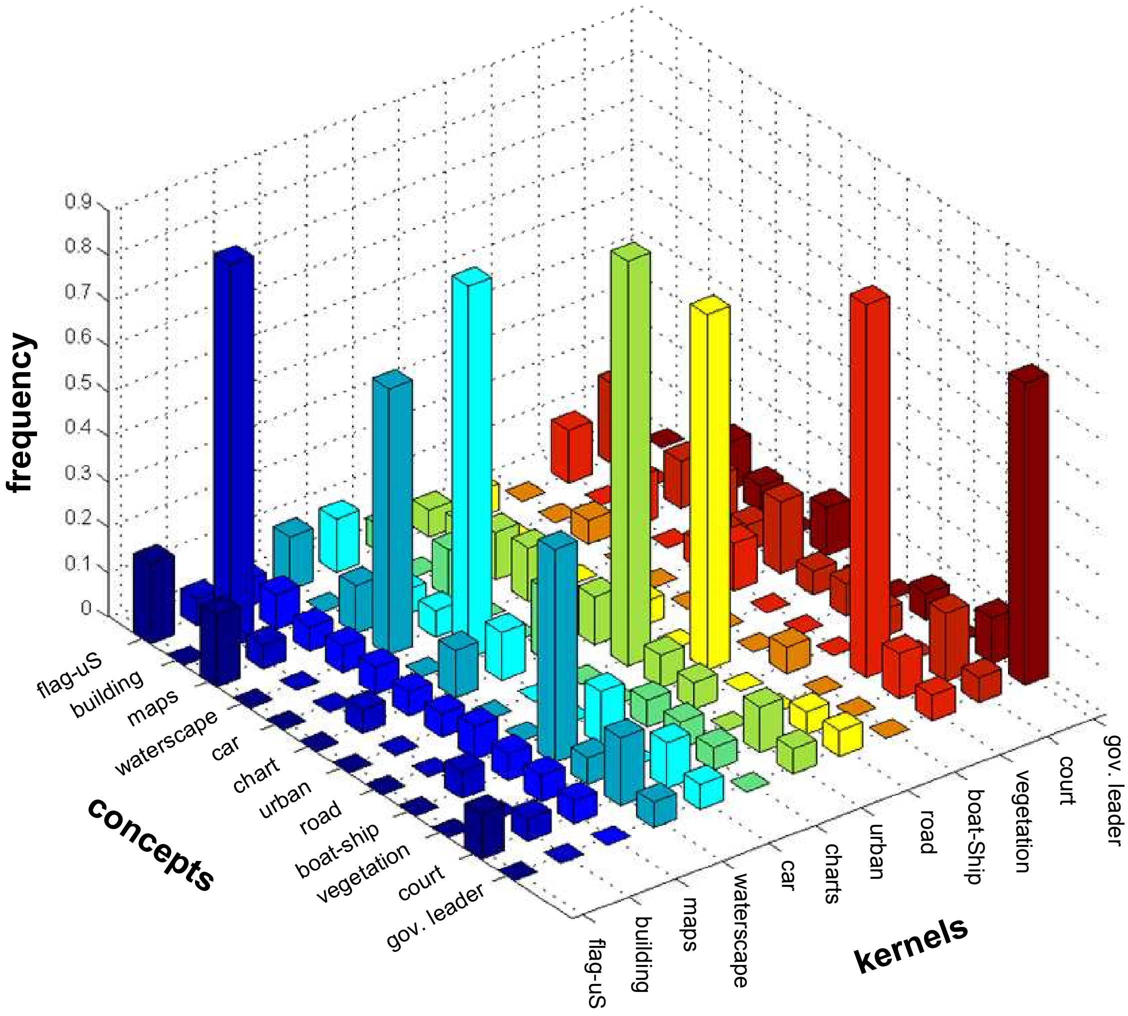


Figure 4.9: Sharing of kernels among different concepts. Each bar gives the frequency a kernel is used in building weak classifiers over 100 iterations for a concept.

train $M \cdot N + \sum_{i=1}^{M-1} i = \frac{(M-1)M}{2} + MN$ classifiers for finding the optimal kernel and subset of concepts, where N is the total size of the kernel pool $\mathbf{K}$. This is larger than the number needed for regular boosting – $M \cdot L$ classifiers during each iteration.

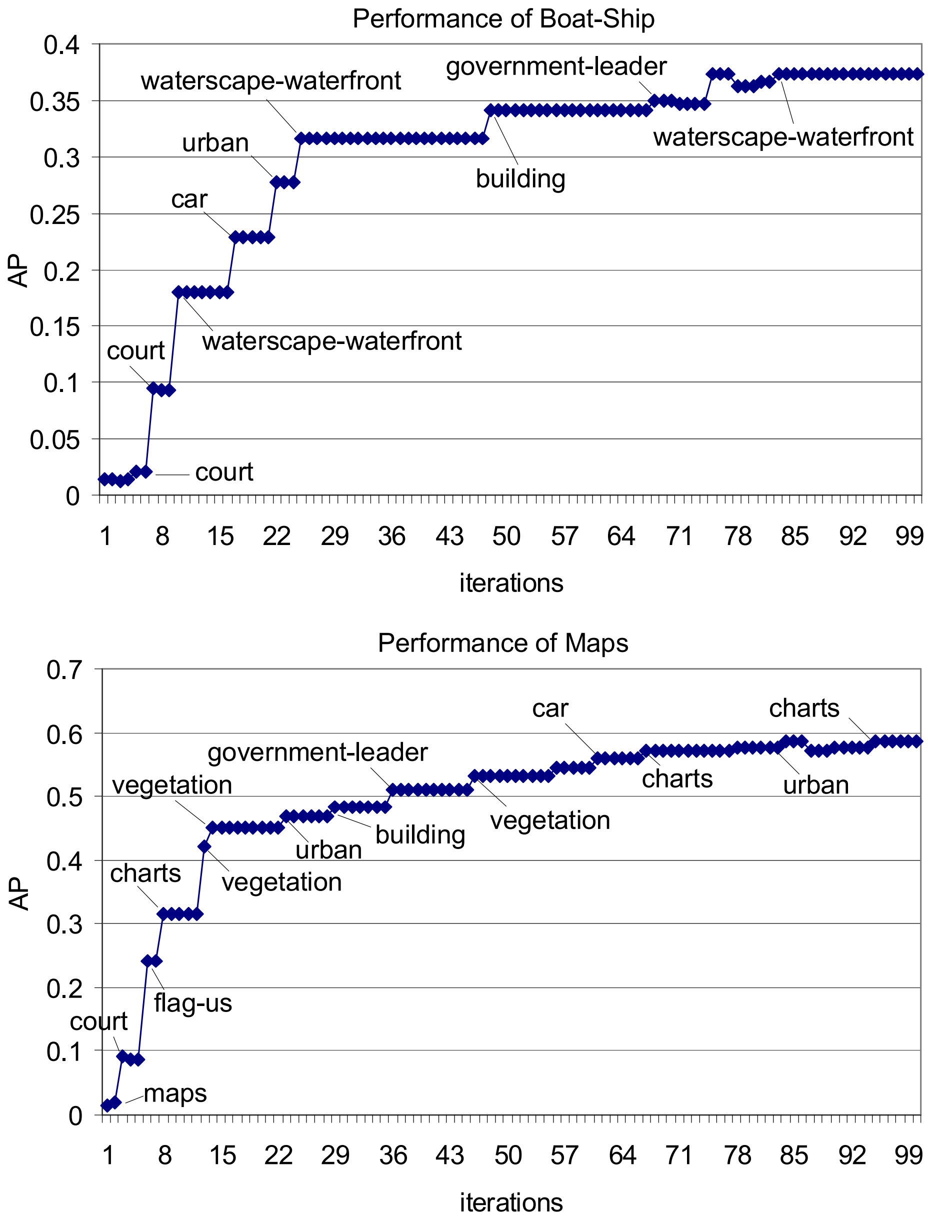


Figure 4.10: Evolution of AP performance through iterations.

4.5 Discussion

In this chapter we propose a novel multi-class concept detection framework based on kernel sharing and joint learning. By sharing "good" kernels among concepts, accuracy of individual detectors can be improved; by joint learning of common detectors across different classes, required kernels and computational complexity for detecting individual concepts can be reduced. We demonstrate our approach by developing an extended JointBoost framework, which automatically selects optimal kernels and subsets of sharing concepts in an iterative boosting process. Our joint boosting method can be applied for multi-class concept detection using any type of kernel. In this work, we propose a VSPM kernel for each individual concept based on the BoF representation. Our method constructs multi-level kernels over multi-resolution visual vocabularies and multi-resolution spatial coordinates. We test our method in detecting 12 concepts over broadcast news videos from the TRECVID 2005 corpus. Clear performance gains are achieved – 10% in MAP and up to 34% AP for some concepts like "maps", "building", and "boat-ship". Extensive analysis of the results also reveals interesting relations among concepts. Our results also confirm the great potential of the proposed direction – leveraging strong visual words constructed from some concepts to help detection of challenging categories.

Chapter 5

Multi-Modal Concept Detection by Direct Fusion

5.1 Introduction

Semantic video concepts are usually manifested through both visual and audio cues. For example, “dancing” is usually associated with human body motions together with background “music”. Each modality brings complementary information about the other and their simultaneous processing can uncover relationships that are otherwise unavailable when considering the signals separately. By intuition, combining evidence from multiple modalities can lead to improved performance. In Section 2.5.3 of Chapter 2, we have introduced many related works using both visual and audio information to help video concept detection, where straightforward multi-modality fusion approach is widely adopted [90, 130, 131, 245]. This is because of the difficulty in object-level visual analysis and audio separation, and the fact that multimodal synchronization may not exist. Such fusion approaches avoid explicit multimodal correlation or synchronization, but directly combine features or classifiers that are separately extracted or trained based on individual modalities.

As illustrated in Fig. 5.1, both early fusion and late fusion can be used to directly combine audio and visual signals in a manner of *coarse synchronization.* That is, some visual feature space $\mathcal{F}^v$ is constructed based on the image keyframe I, and some audio feature space $\mathcal{F}^a$ is generated from the audio soundtrack corresponding to the video segment

u where keyframe I comes from. Then such visual and audio features can be considered as coarsely synchronized since they describe the visual and audio characteristics of the same video segment u. Intuitively, the visual feature vector $\mathbf{f}^v(u)$ (that represents u as a data point in $\mathcal{F}^v$) and the audio feature vector $\mathbf{f}^a(u)$ (that represents u as a data point in $\mathcal{F}^a$) can be directly combined together by either early fusion or late fusion. In early fusion, $\mathbf{f}^v(u)$ and $\mathbf{f}^a(u)$ are concatenated to make a long feature vector to train classifiers. In late fusion, individual classifiers are built for each $\mathbf{f}^v(u)$ and $\mathbf{f}^a(u)$, respectively, and their judgements are combined to make the final decision. The general issues that influence the performance of such direct fusion are the high dimensionality in early fusion, and the classifier combination strategy in late fusion. It is still an open issue how to construct appropriate joint feature vectors comprising of features from different modalities with different time scales, different distance metrics and different dynamic structures. Also, classifier combination remains to be nontrivial [108, 189] and the best combination strategy depends much on the particular problem.

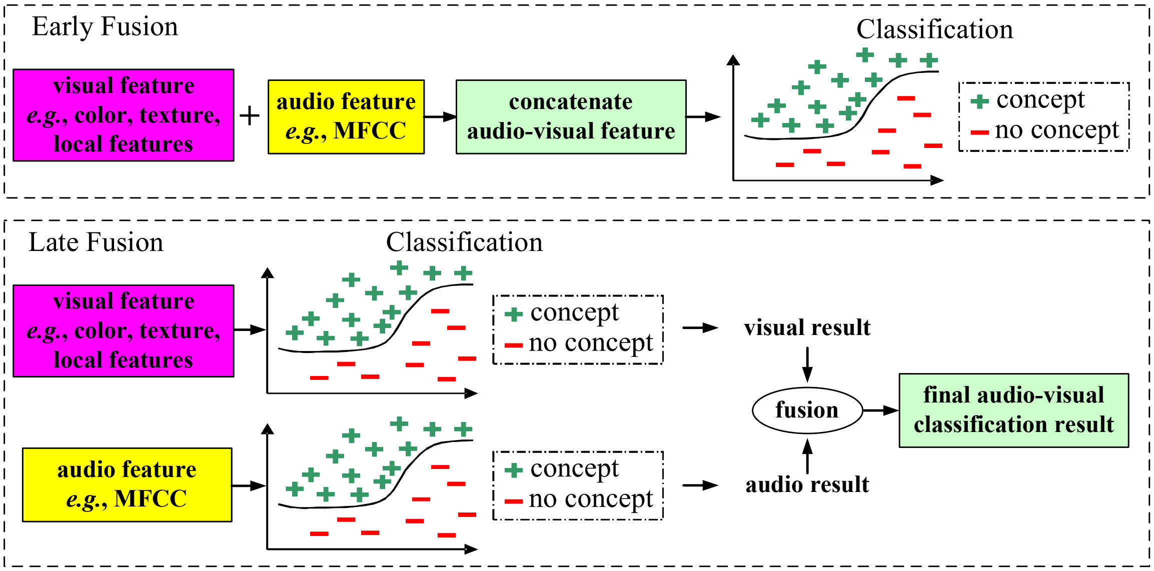


Figure 5.1: Early fusion and late fusion of audio and visual modalities for concept detection.

In this chapter, to exploit the power of both visual and audio cues for concept detection, we develop two multi-modality fusion approaches. We extend the BCRF-CF algorithm proposed in Chapter 3 to an A-V Boosted CRF method, which is a late fusion approach to incorporate both audio and vidual aspects for enhanced classification. Also, we extend the kernel/classifier sharing method proposed in Chapter 4 to an early fusion approach, namely, AV Joint Boosting. Late fusion through A-V Boosted CRF provides a natural way to combine the individual judgements from visual channel and audio channel. Through sharing kernels constructed from multiple modalities with joint boosting, our early fusion A-V Joint Boosting automatically selects the optimal features in classifying different concepts, and can alleviate the problem of "curse of dimensionality". In addition, both A-V Boosted CRF and A-V Joint Boosting can be considered as natural combinations of multi-modality learning and multi-concept learning, and are flexible to incorporate other modalities (*e.g.*, textual) in the future.

We evaluate the proposed multi-modality fusion algorithms over Kodak's consumer benchmark video set [137]. Experimental results show that compared with using individual visual or audio features alone, by combining audio and visual aspects we can get significant performance improvements, *e.g.*, about 10% in terms of MAP. In the following part of this chapter we first introduce an independent audio-based concept detection method, followed by the late fusion A-V Boosted CRF and early fusion A-V Joint Boosting.

5.2 Audio-based Concept Detector

We use the audio classifier we developed in [24]. The soundtracks of each video are described and classified by two techniques, single Gaussian modeling, and pLSA [80] of GMM component occupancy histograms, both described below. All systems start with the same basic representation of the audio, as 25 *Mel-Frequency Cepstral Coefficients* (*MFCCs*) extracted from frequencies up to 7 kHz over 25-ms frames evaluated every 10 ms. Since each video has a different duration, it will result in a different number of feature vectors; these are collapsed into a single video-segment-level feature vector by the two techniques described below. Finally, these fixed-size features are used to compute a similarity kernel matrix (pairwise similarity among a training set consisting of positive and negative samples), and used to train an SVM classifier for each concept.

5.2.1 Single Gaussian Modeling

After the initial MFCC analysis, each soundtrack is represented as a set of $d = 25$ dimensional feature vectors, where the total number depends on the length of the original video. To describe the entire dataset in a single feature vector, we ignore the time dimension and treat the set as samples from a distribution in the MFCC feature space, which we fit with a single 25-dimensional Gaussian by measuring the mean and (full) covariance matrix of the data. This approach is based on common practice in speaker recognition and music genre identification, where the distribution of cepstral features, ignoring time, is found to be a good basis for classification.

To calculate the distance between two distributions, as required in computing the kernel matrix needed in learning the SVM, two approaches were used. One is to use the KL divergence between the two Gaussians. That is, if a video segment u_l has a set of MFCC features, described by a mean vector μ_l and a covariance matrix Σ_l, then the KL distance between video segments u_l and u_k is:

$$D_{KL}(u_l, u_k)^2 = (\mu_l - \mu_k)^T(\Sigma_l^{-1} + \Sigma_k^{-1})(\mu_l - \mu_k) + tr(\Sigma_l^{-1}\Sigma_k + \Sigma_k^{-1}\Sigma_l) - 2d$$

The second approach simply treats the d-dimensional mean vector μ_l concatenated with the $d(d+1)/2$ unique values of the covariance matrices Σ_l as a point in a new (25+325 dimensional) feature space, normalizes each dimension by its standard deviation across the entire training set, and then builds a gram matrix from the Euclidean distance between these normalized feature statistic vectors.

5.2.2 Probabilistic Latent Semantic Analysis

The Gaussian modeling assumes that different activities are associated with different sounds whose average spectral shape, as calculated by the cepstral feature statistics, will be sufficient to discriminate categories. However, a more realistic assumption is that each soundtrack consists of a mixture of many different sounds, each contributing to the sound with a different mixture proportion, leading to variations in the global statistics. If, however, we could decompose the soundtrack into separate descriptions of those specific sounds, we might find that the particular palette of sounds, but not necessarily their exact proportions, would be a more useful indicator of the content. Some sounds (*e.g.*, background noise) may be common to all classes, whereas some sound classes (*e.g.*, a baby's cry) might be very specific to particular classes of video.

To build a model able to capture this idea, we first train a large Gaussian mixture model, containing $M = 256$ Gaussian components, on a subset of MFCC frames chosen randomly from the entire training set. The number of mixtures was optimized in some validation experiments. Such Gaussian mixture model over MFCCs has been successfully used for music classification in previous works [4, 147]. These 256 mixtures are considered as underlying sound classes from which each individual soundtrack is assembled – the analogues of words in document modeling. Then, we classify every MFCC frame in a given soundtrack to one of the mixture components, and describe the overall soundtrack with a histogram of how often each of the 256 Gaussians is present. Note that this representation also ignores temporal structure, but it is able to distinguish nearby points in the cepstral space, depending on how densely a specific neighborhood of feature space is represented, and thus how many Gaussian components it received. The idea of using histograms of acoustic tokens to represent the entire soundtrack is also similar to that used in constructing visual vocabulary histograms for image representation.

We could use this histogram directly, but to remove redundant structure and to give a more compact description, pLSA [80] is further used to summarize the large number of Gaussians into a much more compact representation. This approach, originally developed to model the distributions of words in documents of different topics, models the histogram as a mixture of a smaller number of "topic" histograms, giving each document a compact representation in terms of a small number of topic weights. The individual topics are defined automatically to maximize the ability of the reduced-dimension model to match the original set of histograms. During training, the topic definitions are driven to a local optimum by using the EM algorithm. Specifically, the pLSA representation estimates the probability $p(g|u)$ that a particular component, g, is used in video segment u as the sum of the distributions of for various topic z, $p(g|z)$, weighted by the specific contributions of each topic to video segment u, $p(z|u)$, *i.e.*,

$$p(g|u) = \sum_z p(g|z)p(z|u)$$

The topic profiles $p(g|z)$ (that are shared among all video segments), and the per-video-segment topic weights $p(z|u)$, are optimized by EM. The number of distinct topics determines how accurately the individual distributions can be matched, and also provides a way to smooth over irrelevant minor variations in the use of certain Gaussians. We tuned it empirically on the development data, and found that around 160 topics is the best number for our task. Representing a test item similarly involves finding the best set of weights to match the observed histogram as a combination of the topic profiles; we match in the sense of minimizing the KL distance, which requires an iterative solution. Finally, each video segment is represented by its vector of topic weights, and the SVM's gram matrix (referred to as kernel $\mathcal{K}_{audio}$) is calculated using the Mahalanobis (*i.e.*, covariance-normalized Euclidean) distance in that 160-dimensional space.

5.3 Audio-Visual Boosted CRF

In this section, we will introduce a late fusion framework to combine the audio-based and visual-based concept prediction results from models that are trained separately. The BCRF-CF algorithm is proposed in Chapter 3 as an efficient context-based fusion method for concept detection. This BCRF-CF algorithm has a two-layer framework. In the first layer, independent visual-based concept detectors are applied to get a set of initial posterior probabilities of concept labels on a given image. Then in the second layer the detection results of each individual concept are updated through a context-based model by considering the detection confidence of the other concepts. Here we extend the BCRF-CF method to include models from both visual and audio modalities.

For each video segment u (with corresponding image keyframe I), the input observations are the initial posterior probabilities $\mathbf{h}_I$ generated from concept detectors trained using individual visual and audio features,

$$\mathbf{h}_I = \left[\mathbf{h}_I^{visual}, \mathbf{h}_I^{audio}\right]^T = \left[h_I^{visual,1}, \ldots, h_I^{visual,M}, h_I^{audio,1}, \ldots, h_I^{audio,M}\right]^T$$

where $\mathbf{h}_I^{visual} = \left[h_I^{visual,1}, \ldots, h_I^{visual,M}\right]^T$ includes visual-based independent detection results over every concept C^i ($i = 1, \ldots, M$), and $\mathbf{h}_I^{audio} = \left[h_I^{audio,1}, \ldots, h_I^{audio,M}\right]^T$ includes audio-based independent detection results. Then these inputs are fed into the CRF to get the improved posterior probabilities $P(\mathbf{y}_I|I)$ through inference based on the inter-conceptual relationships. Similar to the original BCRF-CF, A-V Boosted CRF detectors can be learned using the algorithm described in Fig. 3.2 where $\mathbf{h}_I = \left[\mathbf{h}_I^{visual}, \mathbf{h}_I^{audio}\right]^T$.

5.4 Audio-Visual Joint Boosting

Instead of training independent detectors based on visual features and audio features separately, the visual kernels and audio kernels can be jointly used to learn concept detectors. To this end, we adopt the joint boosting and kernel sharing framework developed in the previous Chapter 4 that utilizes a two-stage framework: (1) the kernel construction stage; and (2) the kernel selection and sharing stage. In the first stage, concept-specific kernels such as the VSPM kernels described in Section 4.2, are constructed to capture the most representative characteristics of the visual content for each concept individually. Then in the second stage, these kernels are shared by different concepts through a joint boosting algorithm that can automatically select the optimal kernels from the kernel pool to learn a multi-class concept detector jointly. This two-stage framework can be directly generalized to incorporate audio-based kernels. That is, in the first stage, based on acoustic analysis various kernels can be constructed (such as the audio vocabulary and kernel described in Section 5.2.2), and these kernels can be added into the rich kernel pool together with all the visual-based kernels, and in the second stage the optimal subset of kernels are selected and shared through the joint boosting algorithm.

5.5 Experiments

We evaluate the performance of features, models, and fusion methods described earlier using Kodak's benchmark consumer video set [137]. Among the 25 concepts annotated over the entire video set, we use 21 visual-dominated concepts to evaluate the performance of visual methods and impact of incorporating additional methods based on audio features [1]. The visual-based baseline approach is the SVM classifiers trained using global grid-based color moments, Gabor texture, and edge direction histogram, as described in Section 2.7.4. The audio-based baseline approach is the SVM classifier trained using the global audio feature described in Section 5.2. The A-V Boosted CRF algorithm uses the audio and visual baseline detectors as initial audio and visual independent detectors. The A-V Joint Boosting algorithm shares the SIFT-based VSPM kernels (described in Chapter 4) and the audio kernel (described in Section 5.2) to learn the multi-concept detector.

Each concept detection algorithm is evaluated in five runs and the average performances over all runs are reported. The data sets in the runs are generated as follows: the entire data set $\mathcal{D}$ is randomly split to 5 subsets $\mathcal{D}_1, \ldots, \mathcal{D}_5$ of approximately equal size. By rotating these 5 subsets, we generate the training set, validation set, and test set for each run. That is, for run 1, training set $= \{\mathcal{D}_1, \mathcal{D}_2\}$, validation set $= \mathcal{D}_3$, test set $= \{\mathcal{D}_4, \mathcal{D}_5\}$. Then we switch one subset for run 2, where training set $= \{\mathcal{D}_2, \mathcal{D}_3\}$, validation set $= \mathcal{D}_4$, test set $= \{\mathcal{D}_5, \mathcal{D}_1\}$. Other runs are prepared in a similar fashion. For each run, all algorithms are trained over the training set and evaluated over the test set, except for the A-V Boosted CRF algorithm in which the validation set is used to learn the joint boosting model that fuses individual detectors learned using the training set separately. In addition, we compare our multi-modality fusion methods with a straightforward ensemble method where we simply average independent visual-based and audio-based detection results.

Fig. 5.2 shows the per-concept AP of different audio-visual fusion algorithms, where "All" corresponds to the method that we further average the posteriors from A-V Boosted CRF and A-V Joint Boosting. From the figure, the A-V Boosted CRF algorithm improves the performance of visual baseline by more than 10%. The improvements over many concepts are significant, *e.g.*, 40% over "animal", 51% over "baby", 228% over "museum", 35% over "dancing", and 21% over "parade". These results confirm the power of incorporating inter-concept relations as well as multiple modalities into the fusion model.

[1] The three audio-dominated concepts, *i.e.*, "singing" "music", and "cheer", are not used since we do not have their keyframe-level labels (only labels for the entire videos are available). The "graduation" concept is not used because we do not have enough samples to reasonably evaluate detectors for this concept.

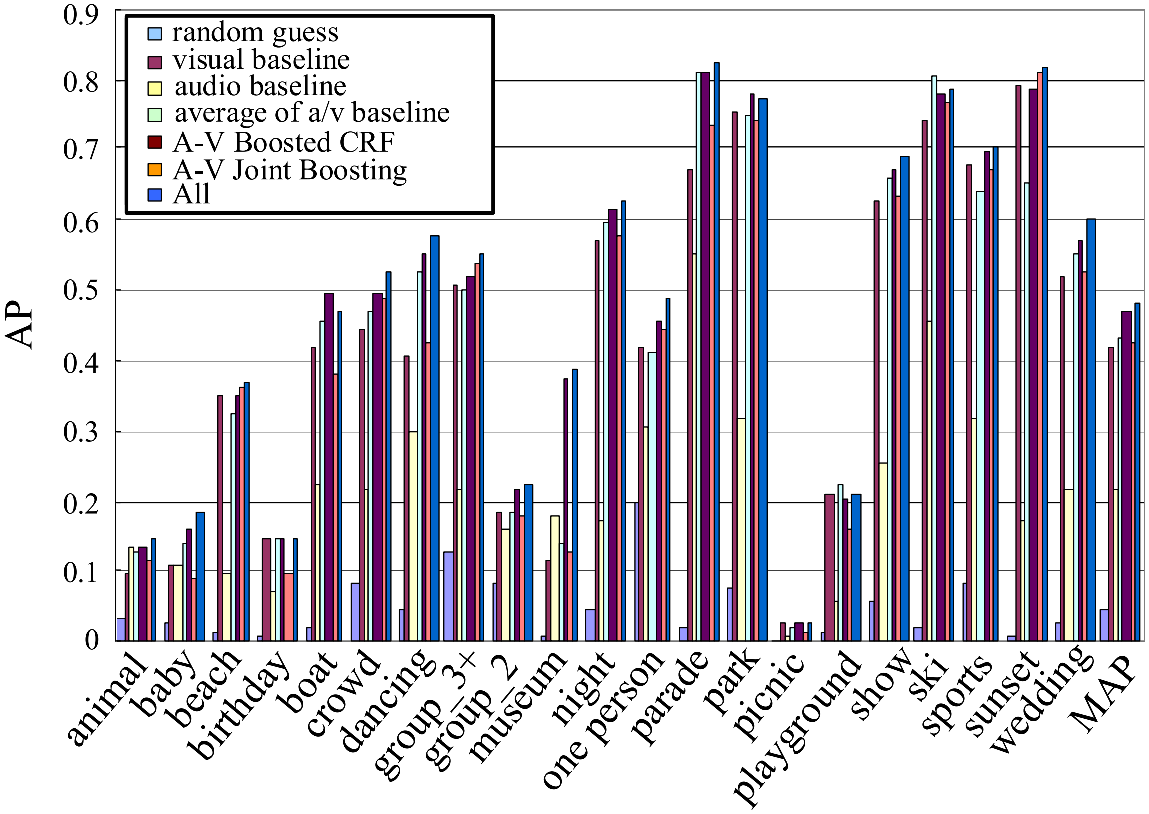
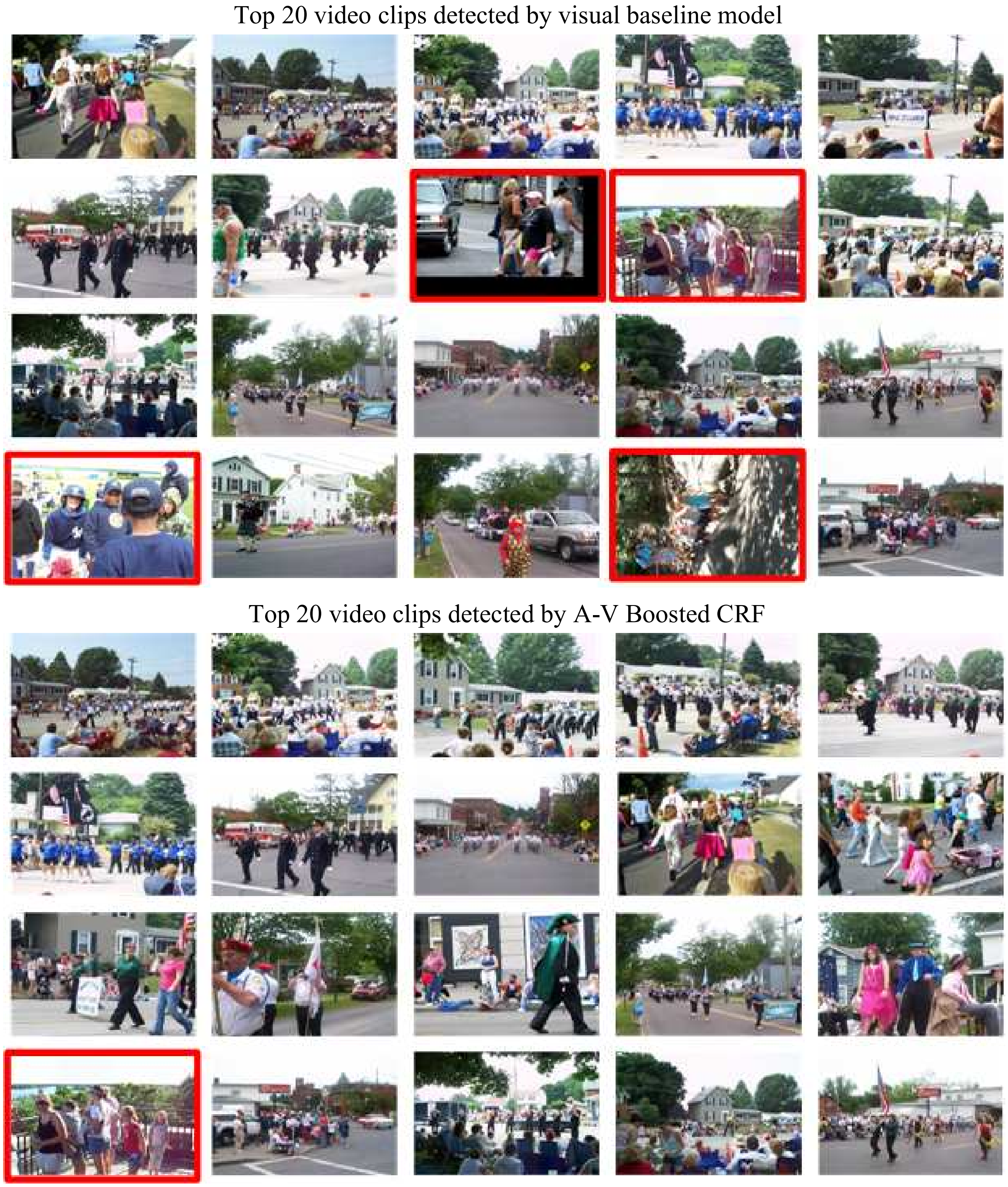


Figure 5.2: Comparison of different audio-visual fusion algorithms.

Compared to straightforward averaging of audio and visual models for each concept, the A-V Boosted CRF context-based fusion method shows more consistent improvement over the diverse set of concepts. Most importantly, it avoids the problem of large performance degradation by averaging models over a few concepts ("sunset" and "museum"), where models from one modality are significantly worse than the others. In other words, by fusing multi-modality models over a large pool of concepts, the stability of the detectors can be greatly improved.

Fig. 5.3 gives an example of the top 20 detected video segments for the "parade" concept (ranked based on the detection scores in descending order) using both A-V Boosted CRF and visual-based baseline. Many irrelevant videos (marked by red rectangles) are included in the top result when using only the visual-based baseline. This is because most of these irrelevant videos contain crowd in the outdoor scene and the visual appearances are similar



Top 20 video clips detected by visual baseline model

Top 20 video clips detected by A-V Boosted CRF

Figure 5.3: Top 20 video segments from the "parade" concept. The irrelevant videos are marked by red rectangles. Video segments are ranked based on the detection scores in descending order.

to those of “parade” keyframes. By using A-V Boosted CRF, such irrelevant videos are removed largely because of the help from the audio models. Parade scenes are usually accompanied by noisy sound from the crowd and loud music. The visual appearances plus audio together can distinguish “parade” videos from visually confusing cases more effectively than only using a single type of feature.

In addition, sharing both audio and visual features among concepts by using A-V Joint Boosting does not result in improved performance on the average. This indicates that the use of audio features and feature sharing in A-V Joint Boosting is not as effective as inter-concept context modeling (via A-V Boosted CRF) for detecting concepts in this data set. This could also be attributed to the difference between local SIFT-based visual features and global visual features in detecting semantic concepts in this Kodak’s consumer video set. However, A-V Joint Boosting does provide complementary benefits – by combining A-V Joint Boosting and A-V Boosted CRF, we achieve further improvements over many concepts, *e.g.*, 10% over “animal”, 12% over “baby”, 7% over “beach”, 7% over “crowd”, 7% over “one person”, *etc.* It is interesting to see that most concepts benefiting from feature sharing (A-V Joint Boosting) overlap with concepts benefiting from context fusion (A-V Boosted CRF). More research is needed to better understand the mechanism underlying this phenomenon, and develop techniques that may automatically discover such concepts.

Analysis of the results from the A-V Joint Boosting models also allows us to investigate the relative contributions of features extracted from videos of individual concepts, and how they are shared across classifiers of multiple concepts. Fig. 5.4 shows the frequency of individual kernels used by the A-V Joint Boosting algorithm in simultaneously detecting 21 concepts through 200 iterations. Only 25 out of the total 64 kernels (3 visual-based kernels for each concept and 1 audio kernel for all concepts) are selected by the feature selection and sharing procedures. It’s surprising to see that single audio kernel turns out to be the most frequently used kernel, more than any other kernels constructed from visual features. This again confirms the importance of multi-modality fusion – despite the lower accuracy achieved by the audio models (compared to their visual counterparts), the underlying audio features play an important role in developing multi-modality fusion models. Furthermore, the feature selection and sharing process used in A-V Joint Boosting is useful in pruning the feature pool in order to make the models more compact. For example, visual kernels computed from training data of “birthday”, “museum”, and “picnic” are discarded because of their relatively poor quality. Images from these concepts have highly diverse visual content and thus the learned visual vocabularies and associated kernels can not capture meaningful characteristics of these concepts.

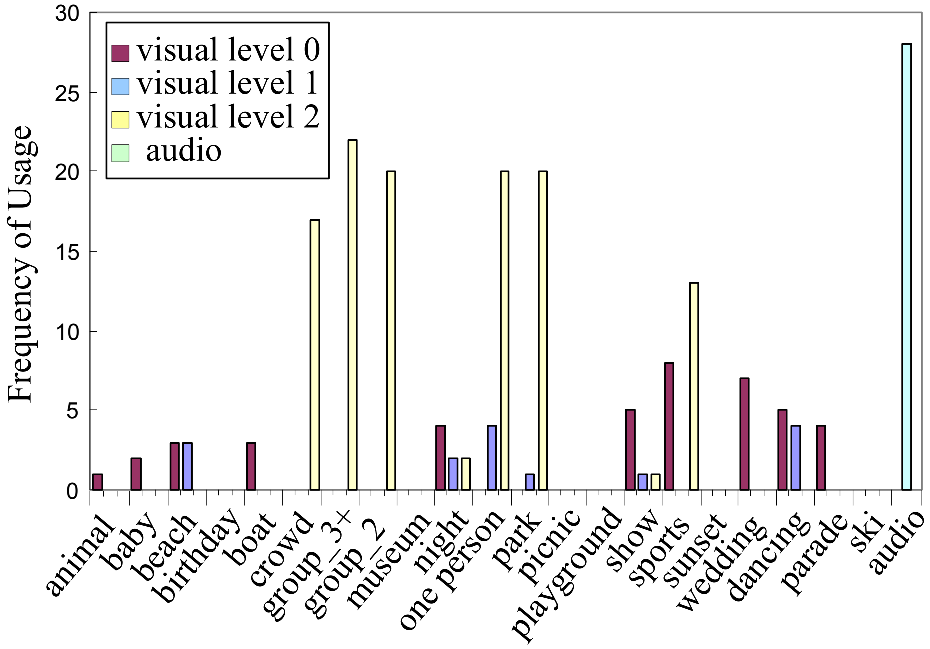


Figure 5.4: Frequency of kernels used by the A-V Joint Boosting algorithm throughout 200 iterations.

5.6 Discussion

In this chapter we develop two multi-modality fusion methods to incorporate both the visual modality and the audio modality for enhanced concept classification. An early fusion approach, A-V Joint Boosting, is proposed to share visual-based and audio-based kernels and classifiers for constructing multi-class concept detectors. An late fusion approach, A-V Boosted CRF, is developed to combine independent audio-based and visual-based concept detection results, by incorporating inter-conceptual relationships. Experiments over Kodak’s consumer benchmark video set demonstrate that both audio and visual features contribute significantly to the robust detection performance by achieving a satisfactory detection accuracy as high as 83% over diverse semantic concepts. The results also confirm the feasibility of semantic classification of consumer videos and suggest interesting future directions.

Chapter 6

Multi-Modality: Mid-Level A-V Atoms

6.1 Introduction

In the previous chapter, we have discussed the multi-modal fusion approaches, where visual features over global image keyframes and audio features from the global audio signal in the same video segment where keyframes come from are directly combined by either early fusion or late fusion to enhance classification. These fusion methods have shown promising results with performance improvements. However, the global visual and audio representations have certain limitation in describing the object-level information, and the disjoint process of extracting audio and visual features limits the ability to generate joint audio-visual patterns that are useful for concept detection. For example, as illustrated in Fig. 6.1, the joint pattern of a birthday cake region and the birthday music is an intuitive strong audio-visual cue for the "birthday" concept but has never been explored in prior works.

On the other hand, as mentioned in Chapter 2, Section 2.5, there are many recent works exploring audio-visual analysis for audio-visual speech/speaker recognition, object localization, object detection and tracking. Such object-based joint audio-visual analysis methods have shown interesting results in analyzing videos in a controlled or simple environment where good foreground/background separation can be obtained in general. However, both object detection and tracking (especially for generic objects) are known to be extremely

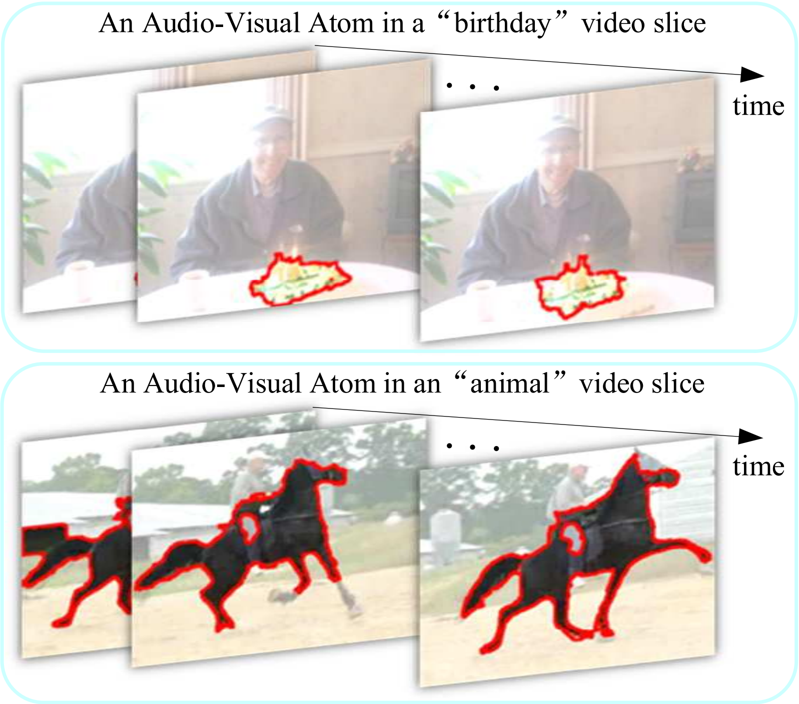

An Audio-Visual Atom in a “birthday” video slice
. . .
time
An Audio-Visual Atom in an “animal” video slice
. . .
time

Figure 6.1: Examples of audio-visual atoms. The region track of a birthday cake and the background birthday music form a salient audio-visual cue to describe “birthday” videos. The region track of a horse and the background horse running footstep sound give a salient audio-visual cue for the “animal” concept.

difficult in general videos. There usually exist uneven lighting, clutter, occlusions, and complicated motions of both multiple objects and camera. In addition, the basic assumption for tight audio-visual synchronization at the object level may not be valid in practice. Multiple objects may make sounds together in a video without large movements, and sometimes the objects making sounds do not show up in the video.

In this chapter, we investigate the challenging issue of joint audio-visual analysis in general videos aiming at detecting generic concepts. We propose a novel representation, the AVA, by extracting atomic representations over video segments. We track automatically segmented regions based on the visual appearance within a short video slice (*e.g.*, 1 second). Regional visual features (*e.g.*, color, texture, and motion) can be extracted from such short-term region tracks. Then based on the visual similarity short-term region tracks from adjacent short-term video slices are connected into long-term region tracks that are called visual atoms. At the same time we locate audio energy onsets from the corresponding audio soundtrack by decomposing the audio signal into the most prominent bases from a time-frequency representation. Audio features (*e.g.*, spectrogram) can be obtained from reconstructed audio signals within a short window around each local energy onset. Such reconstructed short-term audio signals around energy onsets are called audio atoms. Then visual atoms and audio atoms are combined together to form a joint audio-visual atomic representation, *i.e.*, AVAs. Based on AVAs, joint audio-visual codebooks can be constructed, and the codebook-based features can be used for concept detection.

Our method provides a balanced choice for exploring audio-visual correlation in general videos: compared to the previous audio-visual fusion approach in Chapter 5 using coarsely aligned concatenation of global features, we generate an atomic representation in which a moderate level of synchronization is enforced between region tracks and local audio onsets; compared to the tight audio-visual synchronization framework focusing on object detection and tracking, we do not rely on precise object extraction. Compared to alternative methods using static image frames without temporal tracking, less noisy atomic patterns can be found by the short-term tracking characteristics of AVAs. As illustrated by the AVA examples in Fig. 6.1, the temporal region track of a birthday cake associated with the background birthday music gives a representative audio-visual atomic cue for describing "birthday" videos. Similarly, the temporal horse region track together with the horse running footstep sound form a joint audio-visual atomic cue that is salient for describing the "animal" concept. Fig. 6.1 also indicates that the joint audio-visual correlation captured by our AVA is based on co-occurrence, *e.g.*, frequent co-occurrence between a birthday cake and the birthday music. Accordingly, the audio-visual codebooks constructed from salient AVAs can capture the representative audio-visual patterns to describe different individual concepts, and significant detection performance improvements can be achieved.

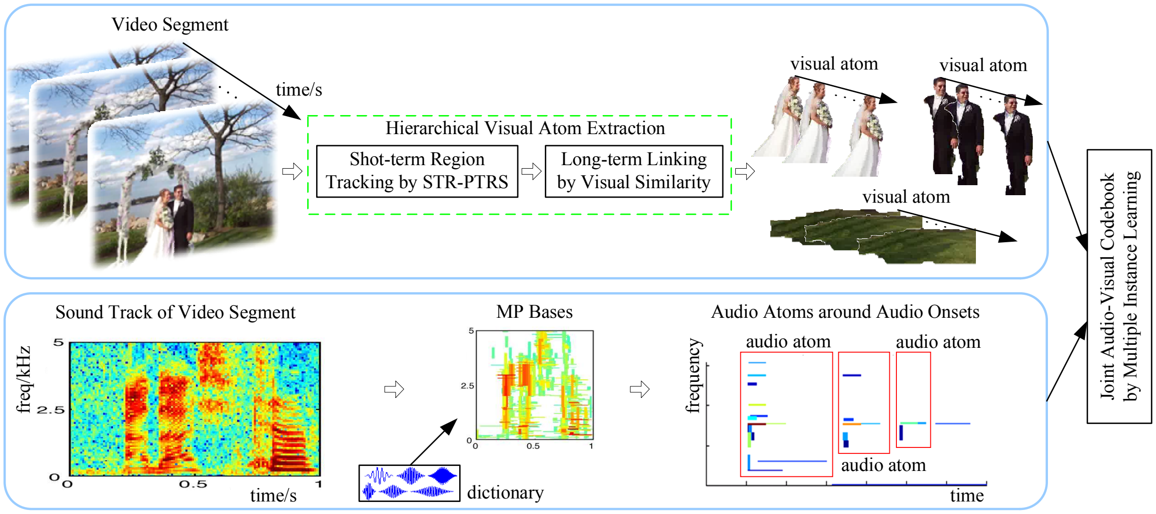

Video Segment
time/s
Hierarchical Visual Atom Extraction
Short-term Region Tracking by STR-PTRS
Long-term Linking by Visual Similarity
visual atom
visual atom
visual atom
Joint Audio-Visual Codebook by Multiple Instance Learning
Sound Track of Video Segment
freq/kHz
5
2.5
0
0
0.5
1
time/s
MP Bases
dictionary
Audio Atoms around Audio Onsets
audio atom
audio atom
audio atom
frequency
time

Figure 6.2: The overall framework of the proposed joint audio-visual analysis approach.

6.2 Overview of Our Approach

Fig. 6.2 shows the framework of our system. We will briefly summarize our work in this section. More details about the visual and audio processes can be found in Section 6.3 and Section 6.4, respectively.

In the visual aspect, we propose a hierarchical framework to extract visual atoms from general videos. In the first step, we develop an effective algorithm, named *Short-Term Region tracking with joint Point Tracking and Region Segmentation* (*STR-PTRS*), to extract short-term region tracks within short windows. STR-PTRS accommodates the challenging conditions in general videos by: conducting temporal tracking within short-term video slices (*e.g.*, 1 second); spatially localizing meaningful regions by image segmentation based on the color and texture appearance; and jointly using interest point tracking and region segmentation to obtain short-term region tracks. The short-term region tracks are not restricted to foreground objects. They can be foreground objects or backgrounds, or combinations of both, all of which carry useful information for detecting various concepts. For example, the red carpet alone or together with the background wedding music are important for classifying the "wedding" concept. Visual features such as color, texture, and spatial location can be generated from the short-term region tracks. Then in the second step, we connect short-term region tracks from adjacent short-term video slices into long-term visual atoms according to the visual similarities calculated using visual features.

With temporal tracking in short-term video slices, better visual atomic patterns can be found compared to the static-region-based alternatives where no temporal tracking is involved. Tracking of robust regions can reduce the influence of noisy regions. Such noise usually comes from imperfect segmentation, *e.g.*, over segmentation or wrong segments due to sudden changes of motion or illumination. By finding trackable visual regions and using such region tracks as whole units to form the final visual atoms, the influence of erroneous segments from a few frames can be alleviated through averaging across the good segments as majorities.

The audio descriptors are based on a *Matching Pursuit* (*MP*) representation of the audio data. MP [146] is an algorithm for sparse signal decomposition from an over-complete set of basis functions, and MP-based audio features have been used successfully for classifying ambient environmental sounds [30]. MP basis functions correspond to concentrated bursts of energy localized in time and frequency and span a range of time-frequency tradeoffs, allowing us to describe an audio signal with the basis functions that most efficiently explain its structure. The sparseness of the representation makes this approach robust to background noise, since a particular element will remain largely unchanged even as the surrounding noise level increases; this is related to a highly robust audio fingerprinting approach explored in [172]. The composition of an MP representation should allow discrimination among the various types of structured (*e.g.* speech and music) and unstructured audio elements that are relevant to concept detection. We extract the audio atoms from the audio soundtrack corresponding to the video window where visual atoms are tracked. Each video window is decomposed into its most prominent elements, and energy onsets are located from the audio signal. Audio atoms are generated as the reconstructed audio sounds within the short windows around energy onsets and are described as spectrogram features.

Visual atoms and audio atoms from the same video window are associated with each other to generate AVAs. Based on the AVA representation, we construct discriminative audio-visual codebooks using the *Multiple Instance Learning* (*MIL*) technique [150] to capture the representative joint audio-visual patterns that are most salient for detecting individual concepts. Precise object detection and tracking are helpful but not required for our approach. This enables us to conduct audio-visual analysis in general videos, different from the previous methods focusing on audio-visual object detection and tracking. We extensively evaluate our algorithm over the challenging Kodak's consumer benchmark video data set from real users [137]. Our method is compared with two state-of-the-art static-region-based image categorization approaches that also use MIL: the DD-SVM algorithm [29] where visual codebooks are constructed by MIL based on static regions and codebook-based features are generated for SVM classification; and the ASVM-MIL algorithm [251] where asymmetrical SVM classifiers are directly built using static regions under the MIL setting. Experiments demonstrate significant improvements achieved by our joint audio-visual codebooks, *e.g.*, over 120% MAP gain (on a relative basis) compared to both DD-SVM and ASVM-MIL. In addition, the joint audio-visual features outperform visual features alone by an average of 8.5% (in terms of AP) over 21 concepts, with many concepts achieving more than 20%.

6.3 Visual Atom Extraction

Detecting and tracking generic objects in general videos are known to be difficult. As can be seen in the example frames from consumer videos in Fig. 2.2 (Chapter 2), there exist dramatic clutter, occlusions, change of shape and angle, and motions of both camera and multiple objects. Most previous tracking algorithms, both blob-based trackers [181, 188, 216, 236] and model-based trackers [8, 71, 94, 185], can not work well. Specifically, blob-based approaches rely on silhouettes derived from variants of background substraction methods, while in general videos due to the complex motions from both objects and camera, and the occlusion of multiple objects, it is very hard to obtain satisfactory silhouettes. On the other hand, most model-based algorithms rely on manual initialization, while for automatic semantic concept detection, such manual initialization is not available. Object detectors can be used to initialize a tracking process [168] but are restricted to tracking some specific objects like human body or vehicle, since it is unrealistic to train a detector for any arbitrary object.

We propose an effective hierarchical framework to extract visual atoms from general videos. In the first step, we temporally track consistent short-term visual regions by an STR-PTRS method. Tracking is conducted within short-term video slices (*e.g.*, 1 second) to accommodate general videos, since only during a short period of time, the changes and movements of the camera and objects are relatively small and there is a high chance to find consistent parts in the frames that can be tracked well. To obtain meaningful regions from a video slice, we use the image segmentation algorithm (that relies on the static color and texture appearance) instead of the background substraction or spatial-temporal segmentation methods [39] that rely on motion. This is because it is very hard to separate camera motion from object motion in general videos and the overall motion is very unstable. In addition, for semantic concept detection not only foreground objects but also backgrounds are useful.

Within each short-term video slice, we jointly use interest point tracking and region segmentation to obtain short-term region tracks. Robust points that can be locked-on well are tracked through the short-term video slice, and based on point linking trajectories, image regions from adjacent frames are connected to generate region tracks. Compared to other possible alternatives, *e.g.*, connecting regions with the similarity over the color and/or texture appearance directly, our approach is more effective in both speed and accuracy: to track a foreground/background region, matching with raw pixel values is not as reliable as matching with robust interest points, due to the change of lighting, shape, and angle; and extracting higher-level color/texture visual features for region matching is quite slow.

The next subsection will describe the detailed STR-PTRS. Now let's formulate our problem. The terminology is consistent with that in Section 2.1. $\mathbf{v}$ denotes a video which is partitioned in to some video segments $u_1, \ldots, u_L$. Each video segment u (we omit index l in the rest subsections without loss of generality since visual atoms are extracted within each video segment independently) is further partitioned into K consecutive short-term video slices $v_1, \ldots, v_K$ (*e.g.*, each v_k has 1-sec length). A set of frames $\tilde{I}_k^1, \ldots, \tilde{I}_k^T$ are uniformly sampled from each video slice v_k with a relatively high frequency, *e.g.*, 30 frames per second or 10 frames per second. Our task is to extract visual atoms from the video segment u.

6.3.1 Short-Term Point Track

Image features (corners *etc.*) that can be easily locked-on are automatically found [207] and then tracked by using the *Kanade-Lucas-Tomasi* (*KLT*) Tracker [13] for every short-term video slice v (again, we omit index k for short-term video slices in this subsection and the next one without loss of generality since short-term region tracks are extracted within each short-term video slice independently). The result is a set of N_p feature tracks, and each feature track has a trajectory $P_j^t = ({x_1}_j^t, {x_2}_j^t)$, where $t = 1, \ldots, T$ is the temporal index (in the unit of frames), j is the index of feature tracks, and x_1, x_2 are the image coordinates.

The KLT tracker is used because of its potent to balance reliability and speed. The KLT tracker defines a measure of dissimilarity that quantifies the change of appearance of a feature between the first and the current image frame, allowing for affine image changes. At the same time, a pure translation model of motion is used to track the selected best features through the sequence. In addition, the maximum inter-frame displacement is limited to improve the reliability and the processing speed. Alternative methods such as tracking with SIFT-based registration [260, 262] generally have limitations in dealing with a large amount of videos (*e.g.*, 1358 videos with 500,000+ frames in our experiments in Section 6.7) due to the speed problem.

In practice, we initiate the KLT tracker with 3000 initial points. In the next subsection, the extracted point tracking trajectories are used to generate short-term region tracks.

6.3.2 Short-Term Region Track

Each frame $\tilde{I}^t$ is segmented into a set of n_r^t homogeneous color-texture regions $r_1^t, \ldots, r_{n_r^t}^t$ by the JSeg tool developed in [40]. Then from each short-term video slice v , we generate a set of N_r short-term region tracks $\mathbf{r}_1, \ldots, \mathbf{r}_{N_r}$ by the algorithm described in Fig. 6.4. Each region track $\mathbf{r}_j$ contains a set of regions $\{r_j^t\}$, where $t = 1, \ldots, T$ is the temporal index (in the unit of frames). The basic idea is that if two regions from the adjacent frames share lots of point tracking trajectories, these two regions are considered as matched regions. To accommodate inaccurate segmentation (where a region from the frame at time t may be separated into several regions at time $t+1$, or several regions from time t may be merged at time $t+1$), we use a replication method to keep all the possible region tracks as illustrated in Fig. 6.3.

Such an approach not only retains all possible region tracks to provide rich information for constructing AVA-based codebooks in later sections, but also helps to reduce the noise from inaccurate segmentation. By treating the short-term region track as a whole unit, the influence of wrong segments from the few frames can be reduced by averaging across good segments as majorities. Finally many replications will have similar visual features and their influences in building the audio-visual codebook will be merged through the further clustering process in later parts of this section.

No further action is taken at this point to fix the inaccurate segmentation. This is because for the purpose of codebook construction in general videos like the ones we deal with there hardly exist universally effective methods in refining the inaccurate segmentation results. However, good refinement of region segments is still favorable and can further improve the quality of the final audio-visual codebook.

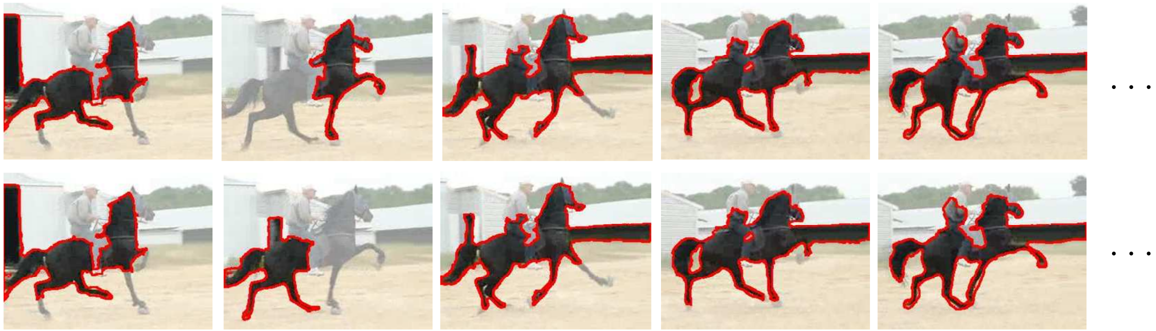


Figure 6.3: An example of region track replication. In the 2^{nd} frame the horse is separated to two parts by inaccurate segmentation. We keep both two possible region tracks.

Note that the above STR-PTRS algorithm may miss some region tracks that enter into the screen in the middle of a short-term video slice. However such regions will still be found in the next video slice as long as they stay in the screen long enough. For those regions that enter and exit the screen very fast (*e.g.*, within a video slice), they are negligible in most general videos for the purpose of semantic concept detection. Similarly, if a shot transition happens within a video slice, most region tracks during the transition may be thrown away, and the final detection performance will hardly be affected. In addition, our STR-PTRS can be extended by adding a backward checking process to overcome this problem. This is also one of our future works.

Input: A set of frames $\tilde{I}^1,\ldots,\tilde{I}^T$ from a short-term video slice v. Regions $r_1^t,\ldots,r_{n_r^t}^t$ for each frame $\tilde{I}^t$. A set of N_p point tracks P_j^t, $j=1,\ldots,N_p$, $t=1,\ldots,T$.

1. Initialization: set $\mathcal{R}=\phi$, $N_r=0$.

2. Iteration: for $t=1,\ldots,T$

- Set $\mathcal{U}=\phi$.
- Calculate:

  $$M_{k,g}^{t|t+1}=\sum_{j=1}^{N_p} I(P_j^t\in r_k^t)I(P_j^{t+1}\in r_g^{t+1})$$

  for each pair of regions $r_k^t\in\tilde{I}^t$, $r_g^{t+1}\in\tilde{I}^{t+1}$.
- For each region $r_k^t\in\tilde{I}^t$:
  - If $M_{k,l^*}^{t|t+1}>H_{low}$ ($l^*=\arg\max_g M_{k,g}^{t|t+1}$), add matched region pair $(r_k^t,\, r_{l^*}^{t+1})$ to $\mathcal{U}$.
  - If $M_{k,l}^{t|t+1}>H_{high}$ ($l\neq l^*$), add matched region pair $(r_k^t,\, r_l^{t+1})$ to $\mathcal{U}$.
- Iteration: for the set of m^t region pairs $(r_k^t, r_{g_1}^{t+1}),\ldots,(r_k^t, r_{g_{m^t}}^{t+1})$ in $\mathcal{U}$ that all start with the same region r_k^t:
  - If there exist m^r region tracks $\mathbf{r}_1,\ldots,\mathbf{r}_{m^r}$, $\mathbf{r}_j\in\mathcal{R}$ that end with r_k^t, replicate each region track $\mathbf{r}_j$ by m^t times, and extend each region track replication by appending $r_{g_1}^{t+1},\ldots,r_{g_{m^t}}^{t+1}$ to the end of each replication respectively. Set $N_r=N_r+m^t\times m^r$.
  - Else, create new tracks $\mathbf{r}_{N_r+1},\ldots,\mathbf{r}_{Nr+m^t}$ starting with r_k^t and ending with each $r_{g_1}^{t+1},\ldots,r_{g_{m^t}}^{t+1}$ respectively. Set $N_r=N_r+m^t$.

3. Remove region tracks in $\mathcal{R}$ with lengths shorter than H_{long}. Output the remaining region tracks.

Figure 6.4: The algorithm to generate short-term region tracks. $I(\cdot)$ is the indicator function. In practice, we empirically set $H_{long}=\frac{1}{2}T$, $H_{low}=10$, $H_{high}=\frac{1}{2}M_{k,l^*}^{t|t+1}$.

To select the appropriate length for short-term video slices, we need to consider two aspects. The video slice needs to be short so that a good amount of point tracking trajectories can be found to get region tracks. On the other hand, the longer the video slice is the better information it retains about temporal movements in visual and audio signals. Fig. 6.5 gives average numbers of point tracking trajectories with changed lengths of video slices. From the figure 1-sec length gives a balanced choice and is used in practice.

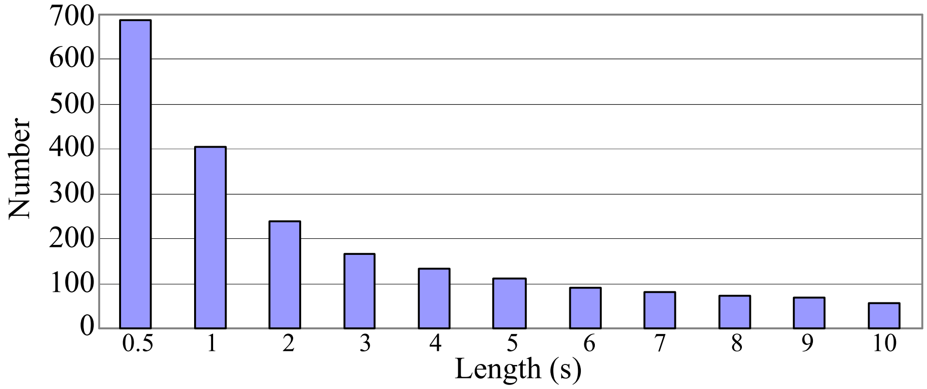


Figure 6.5: Numbers of point tracking trajectories with changing lengths of the short-term video slice. 1-sec length gives a balanced choice and is used in practice.

The appropriate length for short-term region tracking is also related to the degree of content change and motion in the corresponding video. The 1-sec length we take is good for most of the consumer videos we experiment on, which have moderate-level motions. For some special videos, such as sports videos, where lots of fast motions exist, shorter video slices may be needed to capture the fast moving regions. Fig. 6.6 gives some examples of such fast moving objects that can only be captured when we track at 30 fps rate within 0.3-sec video slices. How to design a selection strategy to determine the appropriate length to use according to different videos is one of our future works.

6.3.3 Visual Features over Region Tracks

In this subsection we generate visual feature representations for the short-term region track $\mathbf{r}_j$. First, several types of visual features are extracted from each region $r_j^t \in \mathbf{r}_j$, including color moments in the HSV space (9-dim), Gabor texture (48-dim), and edge direction histogram (73-dim). These features have been shown effective in detecting generic concepts [249]. We concatenate these features into a 130-dim feature vector $\tilde{\mathbf{f}}_{j,vis}^t$ and then average $\tilde{\mathbf{f}}_{j,vis}^t$ across time $t = 1, \dots, T$ to obtain a 130-dim feature vector $\mathbf{f}_{j,vis}$ for the region track $\mathbf{r}_j$. $\mathbf{f}_{j,vis}$ describes the overall visual characteristics of $\mathbf{r}_j$. In addition, optical flow vectors

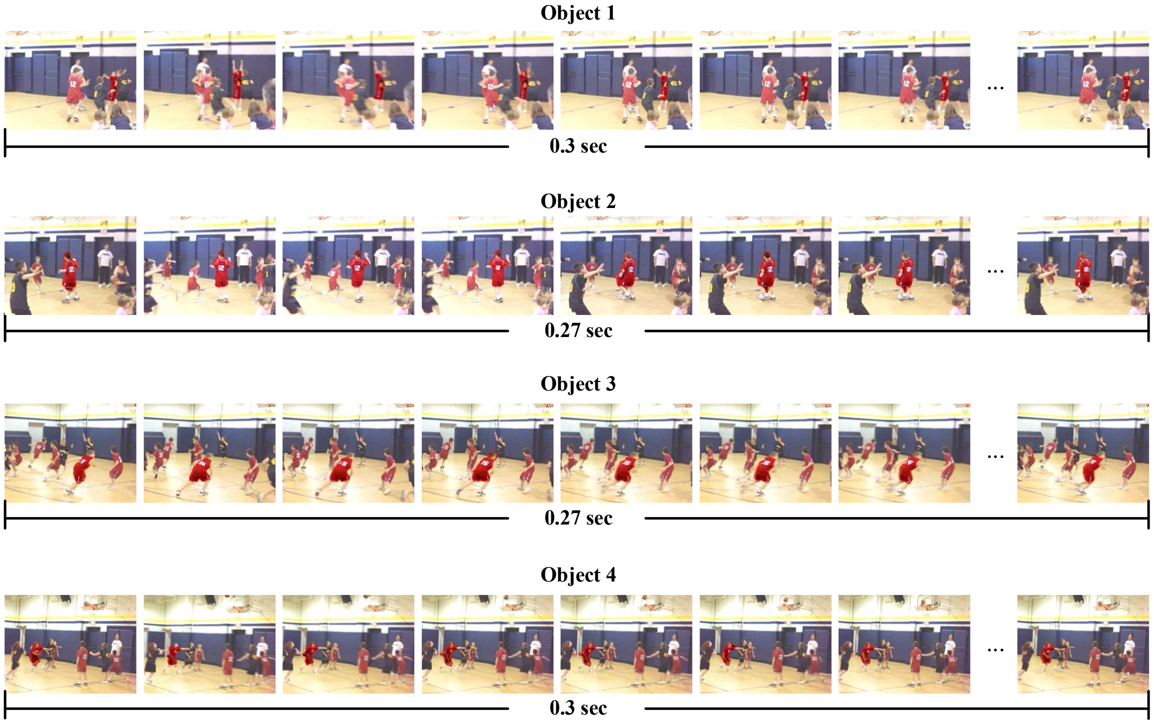


Figure 6.6: Examples of fast moving objects that can only be captured when we track visual regions at 30 fps within 0.3-sec video slices. These objects can not be consistently tracked with 1-sec video slices at 10 fps. The number (*e.g.*, 0.27 sec) below each region track shows the length the track lasts.

are calculated over every pixel of each frame $\tilde{I}_j^t$ using the Lucas-Kanade algorithm [140], where for a pixel $(x1_j^t, x2_j^t)$ a motion vector $[m_1(x1_j^t, x2_j^t), m_2(x1_j^t, x2_j^t)]$ is obtained. Then for each region $r_j^t \in \mathbf{r}_j$, a 4-dim feature vector $\tilde{\mathbf{f}}_{j,mt}^t$ is computed, where each bin corresponds to a quadrant in the 2D motion space and the value for this bin is the average speed of motion vectors moving along directions in this quadrant. For example, the first item in $\tilde{\mathbf{f}}_{j,mt}^t$ is computed as:

$$\frac{1}{R}\sum\nolimits_{(x1_j^t, x2_j^t) \in r_j^t : m_1(x1_j^t, x2_j^t) > 0, m_2(x1_j^t, x2_j^t) > 0} \sqrt{m_1^2(x1_j^t, x2_j^t) + m_2^2(x1_j^t, x2_j^t)}$$

where R is the total size of region r_j^t. Then we average $\tilde{\mathbf{f}}_{j,mt}^t$ across $t = 1, \ldots, T$ to obtain a motion feature vector $\mathbf{f}_{j,mt}$ for the region track $\mathbf{r}_j$. $\mathbf{f}_{j,mt}$ describes the overall moving speed and direction of $\mathbf{r}_j$. The coarse 4-bin granularity is empirically chosen since for the purpose of semantic concept detection fine granularity of motion directions can be very noisy, *e.g.*, an animal can move towards any direction. The coarse description of motion speed and direction gives relatively robust performance in general.

In addition, we generate a spatial feature vector $\mathbf{f}_{j,loc}$ describing the spatial location information for each short-term region track $\mathbf{r}_j$. $\mathbf{f}_{j,loc}$ has three dimensions, corresponding to the horizontal and vertical coordinates of the center of $\mathbf{r}_j$ and the size of $\mathbf{r}_j$, all being the averaged results across the tracked regions in $\mathbf{r}_j$.

Note that more visual features can be extracted to describe short-term region tracks, such as local descriptors like SIFT [138] and HOG [35]. In our experiments, we construct the BoF histogram $\mathbf{f}_{j,sift}$ for the region track $\mathbf{r}_j$ based on a codebook generated by clustering SIFT features from a set of training videos, following the recipe of [64]. These SIFT descriptors are calculated over interest points detected with the Hessian-Affine algorithm [153]. However, for concept detection over our consumer videos, in general local SIFT can not compete with the global regional visual feature $\mathbf{f}_{j,vis}$, as will be demonstrated in the experiments. This phenomenon intuitively confirms how challenging this consumer collection is. Due to the large diversity in the visual content, there is very little repetition of objects or scenes in different videos, even those from the same concept class. In such a case, it is hard to exert the advantage of local descriptors like SIFT for local point matching/registration. Nonetheless, local features can still be used as additional descriptors to complement the global regional visual features.

6.3.4 Long-term Linking to Generate Visual Atoms

By now, for an input video segment u, over each short-term video slice v_k, $k = 1, \ldots, K$, we have a set of N_r^k short-term region tracks $\mathbf{r}_1^k, \ldots, \mathbf{r}_{N_r^k}^k$. In this subsection, we connect these short-term region tracks from adjacent video slices together into long-term visual atoms. Such linking is based on the low-level visual feature $\mathbf{f}_{j,vis}^k$ and the location information $\mathbf{f}_{j,loc}^k$ of each region track $\mathbf{r}_j^k$. Fig. 6.7 gives the algorithm used for long-term linking.

Fig. 6.8 shows some example visual atoms generated after long-term linking. From the examples we can see that over some consistent large regions we can get pretty good visual atoms extracted that capture salient regions in general videos. Actually, over the ex-

```
Input: A set of short-term video slices $v_1, \dots, v_K$ from an input video segment $u$. A set of $N_r^k$ short-term region tracks $\mathbf{r}_1^k, \dots, \mathbf{r}_{N_r^k}^k$ for each short-term video slice $v_k$. Visual feature $\mathbf{f}_{j,vis}^k$ and spatial feature $\mathbf{f}_{j,loc}^k$ associated with each region track $\mathbf{r}_j^k$.

1. Initialization: set $\mathcal{R}=\phi$, $N_{vis}=0$.

2. Iteration: for $k=1,\dots,K$
    • Set $\mathcal{U}=\phi$.
    • Calculate the pairwise distance $D_{vis}(\mathbf{r}_i^k, \mathbf{r}_j^{k+1})$ and $D_{loc}(\mathbf{r}_i^k, \mathbf{r}_j^{k+1})$ between region tracks $\mathbf{r}_i^k \in v_k$ ($i = 1, \dots, N_r^k$) and region tracks $\mathbf{r}_j^{k+1} \in v_{k+1}$ ($j = 1, \dots, N_r^{k+1}$) over the visual feature $\mathbf{f}_{vis}$ and spatial feature $\mathbf{f}_{loc}$, respectively.
    • For each region track $\mathbf{r}_i^k \in v_k$:
        – If $D_{vis}(\mathbf{r}_i^k, \mathbf{r}_{l^*}^{k+1}) < H_{vis}^{low}$ & $D_{loc}(\mathbf{r}_i^k, \mathbf{r}_{l^*}^{k+1}) < H_{loc}$ (where $l^* = \arg\min_j D_{vis}(\mathbf{r}_i^k, \mathbf{r}_j^{k+1})$), add matched pair of region tracks $(\mathbf{r}_i^k, \mathbf{r}_{l^*}^{k+1})$ to $\mathcal{U}$.
        – If $D_{vis}(\mathbf{r}_i^k, \mathbf{r}_l^{k+1}) < H_{vis}^{high}$ & $D_{loc}(\mathbf{r}_i^k, \mathbf{r}_l^{k+1}) < H_{loc}$ ($l \neq l^*$), add matched pair of region tracks $(\mathbf{r}_i^k, \mathbf{r}_l^{k+1})$ to $\mathcal{U}$.
    • Iteration: for the set of $m^k$ pairs of region tracks $(\mathbf{r}_j^k, \mathbf{r}_{g_1}^{k+1}), \dots, (\mathbf{r}_j^k, \mathbf{r}_{g_{m^k}}^{k+1})$ in $\mathcal{U}$ that all start with the same region track $\mathbf{r}_j^k$:
        – If there exist $m^a$ visual atoms $\mathcal{A}_1^{vis}, \dots, \mathcal{A}_{m^a}^{vis}$, $\mathcal{A}_i^{vis} \in \mathcal{R}$ that end with region track $\mathbf{r}_j^k$, replicate each visual atom $\mathcal{A}_i^{vis}$ by $m^k$ times, and extend each visual atom replication by appending region tracks $\mathbf{r}_{g_1}^{k+1}, \dots, \mathbf{r}_{g_{m^k}}^{k+1}$ to the end of each replication respectively. Set $N_{vis} = N_{vis} + m^k \times m^a$.
        – Else, create new visual atoms $\mathcal{A}_{N_{vis}+1}^{vis}, \dots, \mathcal{A}_{N_{vis}+m^k}^{vis}$ starting with region track $\mathbf{r}_j^k$ and ending with each $\mathbf{r}_{g_1}^{k+1}, \dots, \mathbf{r}_{g_{m^k}}^{k+1}$ respectively. Set $N_{vis} = N_{vis} + m^k$.

3. Output the visual atoms in $\mathcal{R}$.
```

Figure 6.7: The algorithm for long-term linking to generate visual atoms. We empirically set H_{vis}^{high}=0.25, H_{vis}^{low}=0.2, H_{loc}=0.15$|Ig|$ where $|Ig|$ is the size of a video frame. Note that parameters H_{vis}^{high}, H_{vis}^{low}, and H_{loc} may change according to different types of videos.

ample "wedding" video, there are over 100 visual atoms extracted in total, and in the figure we only show a few of them. About 80% of the remaining visual atoms are relatively short (*e.g.*, lasting for two to three seconds). This is due to the dramatic content change and motion in such general consumer videos.

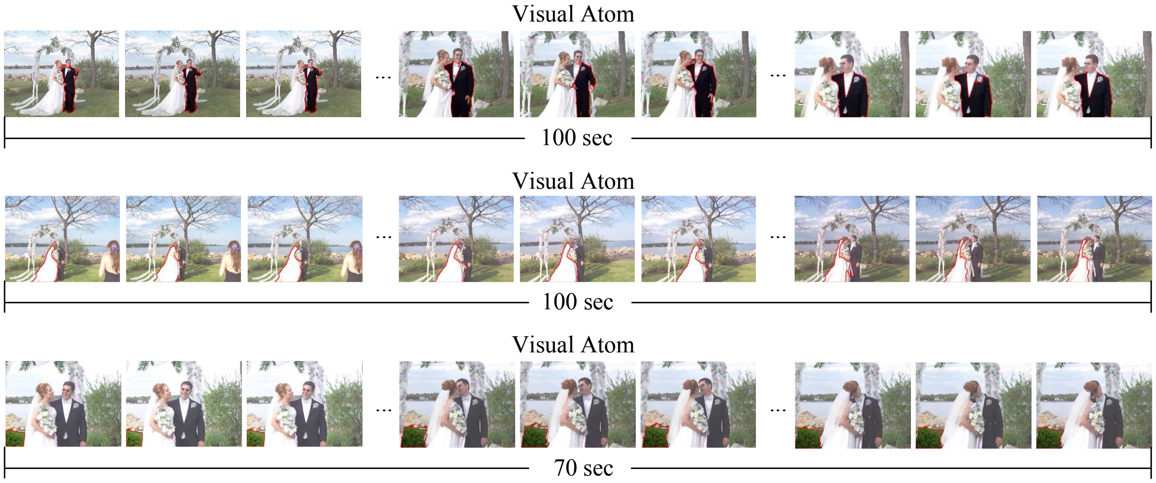


Figure 6.8: Examples of the final visual atoms extracted from a "wedding" video. There are over 100 visual atoms extracted in total, and here we only show a few example. About 81% of the remaining visual atoms are relatively short (*e.g.*, lasting for two to three seconds), due to the dramatic content/motion change in such general consumer videos.

6.3.5 Refinement through Clustering

For each input video segment u, we have a set of N_{vis} visual atoms extracted as $\mathcal{A}_1^{vis}, \ldots, \mathcal{A}_{N_{vis}}^{vis}$. Each visual atom corresponds to a set of visual regions that are spatially continuous with consistent visual appearances through time. The visual atoms can capture individual objects, parts of objects, or combinations of multiple objects, through time. However, such representation is not temporally complete, i.e., a visual atom may only capture a consistent region over discontinued time windows. As illustrated in Fig. 6.9, we track the groom's face for a while, then the track is lost, and later on we track his face for another while again. Such incomplete tracking results can be attributed to the challenging conditions for visual tracking in general videos. In this subsection, we refine the extracted raw visual atoms through a clustering process, where we try to group visually similar visual atoms together as one entity for later usage. The goal is that such clustered groups can merge visual atoms that describe the same visual regions (but are tracked separately at different times) together. In practice, the hierarchical clustering algorithm is used to cluster visual atoms according to visual features $\mathbf{f}_{vis}$. Fig. 6.10 shows some examples of the clustering results. As shown in the figure, such similar visual atoms can be consistent regions separated by inconsistent tracking or similar regions resulting from over segmentation. This process has the effect of de-noising the visual atoms and removing the redundant visual atoms.

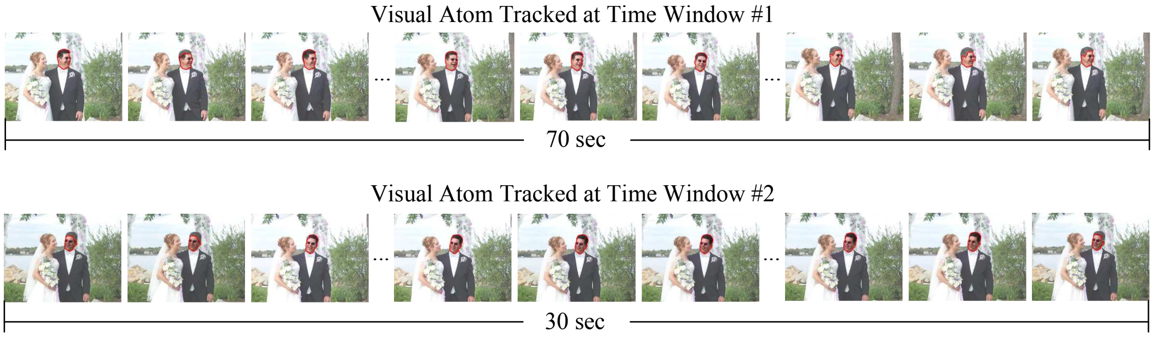

Visual Atom Tracked at Time Window #1
...
...
70 sec
Visual Atom Tracked at Time Window #2
...
...
30 sec

Figure 6.9: Illustration of incomplete visual tracking. The visual atoms may only capture a consistent region over discontinued time windows.

6.4 Audio Atom Extraction

In this section, we describe the process to extract audio atoms from the sound track corresponding to each video segment u where visual atoms are generated.

6.4.1 Extracting Time-Frequency Bases

We represent the audio sound using a matching pursuit decomposition [146]. The bases used for MP are Gabor functions, which are Gaussian-windowed sinusoids. The Gabor function is evaluated at a range of frequencies covering the available spectrum, scaled in length (trading time resolution for frequency resolution), and translated in time. The created functions form a dictionary, which possesses a continuum of time-frequency localization properties.

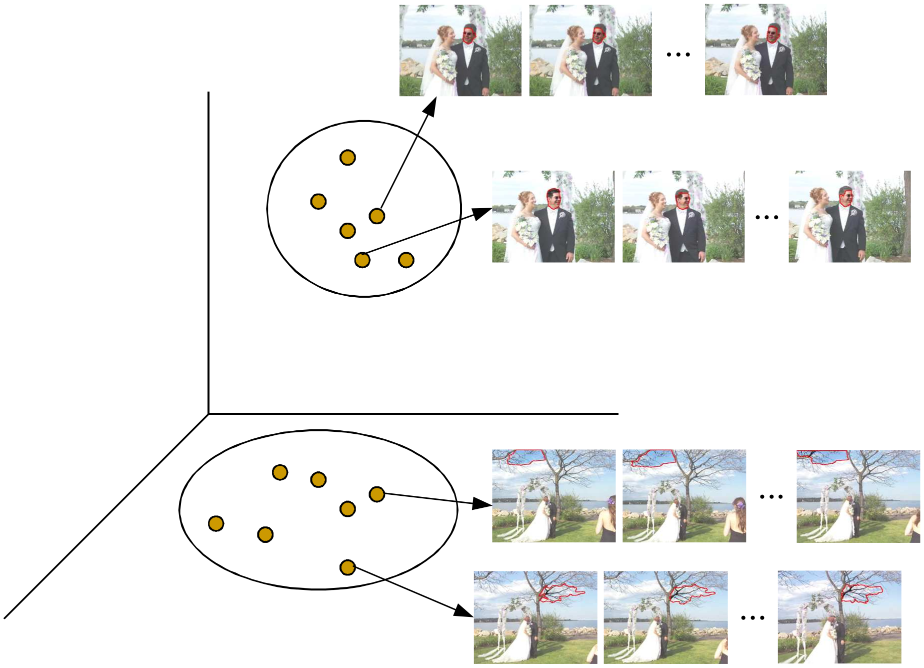


Figure 6.10: Examples of clustered visual atoms. Through clustering we can group similar visual atoms together, including consistent regions separated by inconsistent tracking and similar regions resulting from over segmentation.

The length scaling creates long functions with narrowband frequency resolution, and short functions (well-localized in time) with wideband frequency resolution. This amounts to a modular *Short-Time Fourier Transform* (*STFT*) representation, with analysis windows of variable length. During MP analysis, functions are selected in a greedy fashion to maximize the energy removed from the signal at each iteration, resulting in a sparse representation. This process has the effect of de-noising the signal while retaining information about the most important parts of the signal. The Matching Pursuit Toolkit [66], an efficient implementation of the algorithm, is used. The dictionary contains functions at eight length scales, incremented by powers of two. For data sampled at 16 kHz, this corresponds to durations ranging from 2 to 256 ms. These are each translated in increments of one eighth of the function length, over the duration of the signal.

To ensure coverage of the audio activity in each short-term window, we extract a fixed number of functions (500) from each window. We then prune this set of functions with post-processing based on psychoacoustic masking principles [186]. This emulates the perceptual effect by which lower energy functions close in frequency to higher-energy signal cannot be detected by human hearing. We retain the 70% of the functions with the highest perceptual prominence relative to their local time-frequency neighborhood. This emphasizes the most salient functions, and removes less noticeable ones. Fig. 6.11 shows the original audio signal and the extracted time-frequency bases.

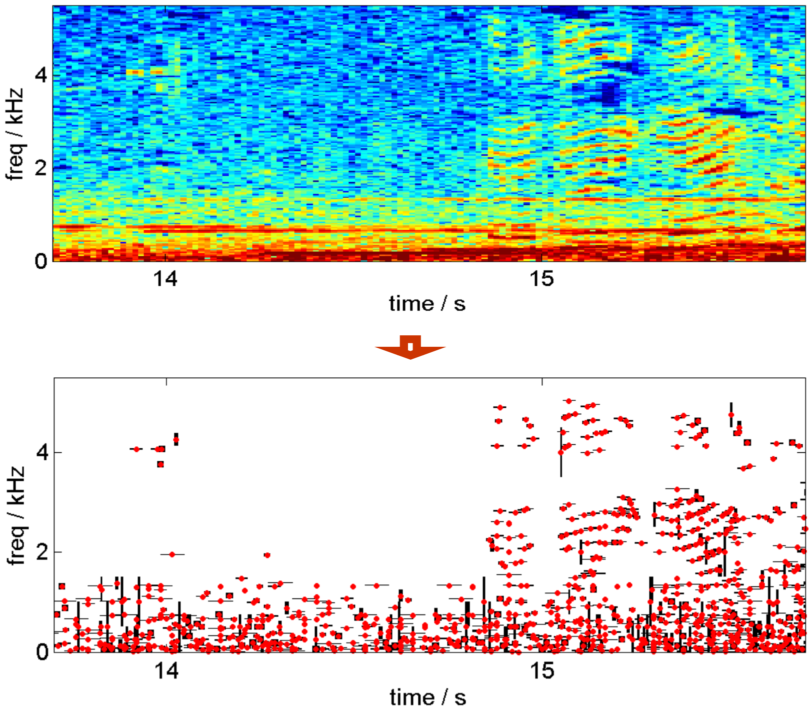


Figure 6.11: An example of the original audio signal and the corresponding extracted time-frequency bases.

6.4.2 Locate Audio Onsets

Using a detection function built from the positions, lengths, and amplitudes of the extracted time-frequency bases, we determine a set of times at which we think an onset of energy probably occurred in the audio. To do this we sum the envelopes of all the functions that are retained after the pruning process; this is our onset detection function. We then use two different approaches to select peaks from this function. First we select all peaks from the detection function and keep those that are above a threshold (some percentage above the local mean). This detects most sharp onsets. To detect softer but still significant onsets, we smooth the original detection function by low-pass filtering, and then repeat the peak-picking process. The two sets of potential onset times are then combined. Additionally, we prune the final set of onset times by requiring that onsets be no closer together than 50 ms. We therefore remove small peaks that are closer to a larger peak than 50 ms.

6.4.3 Audio Atoms and Features

For each energy onset, we collect all those time-frequency bases whose center times fall within a short window around the onset time, and we throw away all other time-frequency bases from the audio signal. Then we reconstruct the audio signal around each onset separately, using only those remaining few time-frequency bases. After that, we generate a coarsely-binned mel-frequency spectrogram representation of the short time window around each onset. Compared to conventional features like MFCCs, these new features are designed to be relatively invariant to background noise and to variations in acoustic channel characteristic, due to the focus on energy onsets. Such audio features around energy onsets provide a natural domain for segmenting the representation into portions associated with distinct objects. This ability gives the opportunity to study moderate-level audio-visual correlation.

6.4.4 Refinement through Clustering

We perform PCA on the audio spectrograms collected from all onsets from all training videos and keep the first 20 principle components. Then for each video segment u, we cluster the extracted audio atoms in the 20-dim PCA space. The centers of the few largest clusters are used as the final audio atoms to correlate with the visual atoms. Fig. 6.12 shows some examples of the audio spectrograms over reconstructed audio signals around onsets. It also illustrates the clustering process to obtain the refined audio atoms.

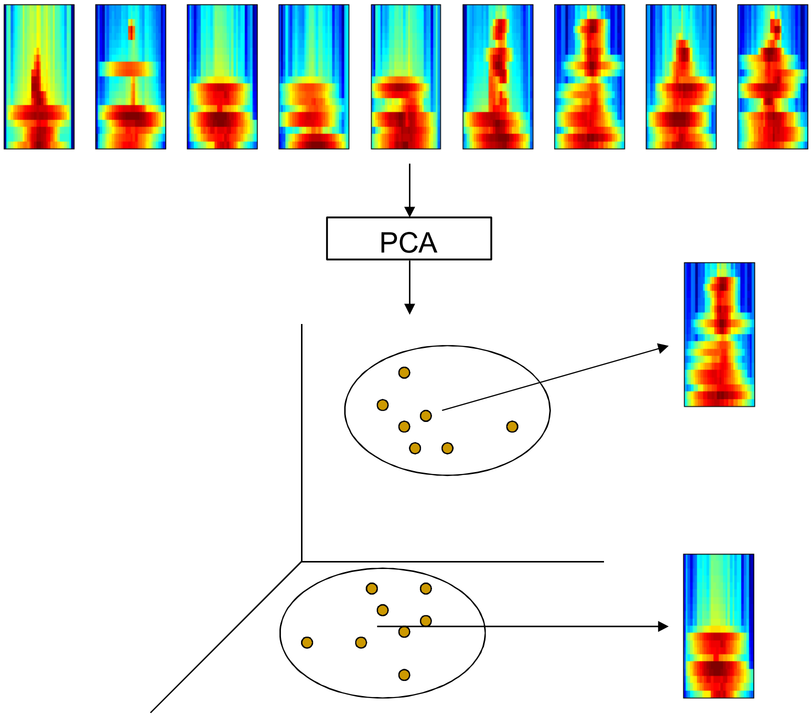
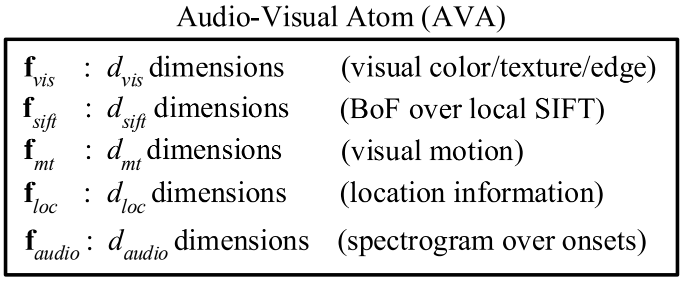


Figure 6.12: Example of clustered audio atoms.

6.5 Joint Audio-Visual Codebook Construction

So far for each input video segment u, we have a set of N_{vis} visual atoms $\mathcal{A}_1^{vis}, \ldots, \mathcal{A}_{N_{vis}}^{vis}$ and a set of N_{aud} audio atoms $\mathcal{A}_1^{aud}, \ldots, \mathcal{A}_{N_{aud}}^{aud}$. We associate each audio atom with each visual atom and generate a cross-product number of AVAs: $\mathcal{A}_i^{vis-aud}$, $i = 1, \ldots, N_{vis} \times N_{aud}$. As illustrated in Fig. 6.13, each AVA $\mathcal{A}_i^{vis-aud}$ contains a long-term region track associated with a global visual feature vector $\mathbf{f}_{i,vis}$ (d_{vis} dimensions), a local SIFT feature vector $\mathbf{f}_{i,sift}$ (d_{sift} dimensions), a spatial feature vector $\mathbf{f}_{i,loc}$ (d_{loc} dimensions), a motion feature vector $\mathbf{f}_{i,mt}$ (d_{mt} dimensions), and an audio feature vector $\mathbf{f}_{i,audio}$ (d_{audio} dimensions). We can concatenate different types of features into various multi-modal vectors, based on which different multi-modal codebooks can be constructed.

As described in Chapter 2, Fig. 2.1, a video concept detection task usually has the following formulation: keyframes are sampled from each video segment and are annotated with binary labels. In our experiment, one keyframe I_l is sampled from each video segment



Audio-Visual Atom (AVA)

$\mathbf{f}_{vis}$	: d_{vis} dimensions	(visual color/texture/edge)
$\mathbf{f}_{sift}$	: d_{sift} dimensions	(BoF over local SIFT)
$\mathbf{f}_{mt}$	: d_{mt} dimensions	(visual motion)
$\mathbf{f}_{loc}$	: d_{loc} dimensions	(location information)
$\mathbf{f}_{audio}$	: d_{audio} dimensions	(spectrogram over onsets)

Figure 6.13: Structure of the AVAs in our implementation. This structure can be easily extended to accommodate other types of features.

u_l. A binary label $y^m_{I_l}\!=\!1$ or -1 is assigned to each keyframe I_l to indicate the occurrence or absence of a concept C^m ($m = 1, \ldots, M$) in the video segment u_l. Based on this structure, we extract a set of AVAs over each video segment, and then we use the extracted AVAs to construct a discriminative joint audio-visual codebook for each concept C^m.

Each video segment u_l can be treated as a "bag-of-AVAs", *i.e.*, it consists of a set of AVAs generated from the previous sections, and each AVA is an instance in the video-segment bag. Thus y_I is the label over the bag rather than over instances. For a semantic concept C^m, it is sensible to assume that a "positive" bag u_l (with $y^m_{I_l} = 1$) must have at least one of its instances being "positive", *e.g.*, a positive video segment for concept "animal" must have at least one "animal" AVA. On the other hand, a "negative" bag u_l (with $y^m_{I_l} = -1$) does not have any "positive" instance. This formulation is known as MIL [29, 150, 251] in the literature.

With different concatenations of $\mathbf{f}_{i,vis}$, $\mathbf{f}_{i,loc}$, $\mathbf{f}_{i,sift}$, $\mathbf{f}_{i,mt}$, and $\mathbf{f}_{i,audio}$, various multi-modal features can be generated to describe an AVA $\mathcal{A}^{vis-aud}_i$. Assume that we have a combined d-dim feature space. For each concept C^m, we repeat an MIL-type procedure P_m-times in order to obtain P_m discriminative prototypes $(\mathbf{f}^{m*}_p, \mathbf{w}^{m*}_p)$, $p\!=\!1, \ldots, P_m$, consisting of a prototype point (or centroid) $\mathbf{f}^{m*}_p = [f^{m*}_{p1}, \ldots, f^{m*}_{pd}]^T$ in the d-dim feature space, and the corresponding weights for each dimension $\mathbf{w}^{m*}_p\!=\![w^{m*}_{p1}, \ldots, w^{m*}_{pd}]^T$.

Among the flavors of MIL objective functions, the *Diverse Density* (*DD*) is one that fits our intuitive objective above and also with efficient inference algorithm available [29] via EM. In the rest of Section 6.5, we omit subscripts m, p without loss of generality, as each $\mathbf{f}^*$ will be independently optimized for different concepts over different video segment bags $l \in \{1, \ldots, L\}$ and different instances $j \in \{1, \ldots, N_l\}$ in each bag u_l. The DD objective function for one bag u_l can be simply written as:

$$Q_l = \frac{1 + y_{I_l}}{2} - y_{I_l} \prod\nolimits_{j=1}^{N_l} (1 - e^{-||\mathbf{f}_{lj} - \mathbf{f}^*||^2_{\mathbf{w}^*}}) \tag{6.1}$$

where $\mathbf{f}_{lj}$ is the feature vector of the j-th AVA instance, and $||\mathbf{f}||_{\mathbf{w}}$ is the weighted 2-norm of vector $\mathbf{f}$ by $\mathbf{w}$, *i.e.*, $||\mathbf{f}||_{\mathbf{w}} = (\sum_{i=1}^{d} (f_i w_i)^2)^{\frac{1}{2}}$. For a positive bag u_l, Q_l will be close to 1 when $\mathbf{f}^*$ is close to any of its instances, and Q_l will be small when $\mathbf{f}^*$ is far from all its instances. For a negative bag u_l, Q_l will be large when $\mathbf{f}^*$ is far from all its instances. By aggregating Eqn. (6.1) over all bags the optimal $\mathbf{f}^*$ will be close to instances in the positive bags and far from all of the instances in the negative bags.

For each positive video segment bag u_l, there should be at least one AVA to be treated as a positive sample to carry the label of that bag. This instance, denoted by $J(u_l)$, is identified as the closest instance to the prototype $\mathbf{f}^*$ and is given by Eqn. (6.2). For each negative bag u_l (with $y_{I_l} = -1$), on the other hand, all instances are treated as negative samples, whose contributions to Q_l are all preserved.

$$J(u_l) = \arg\max\nolimits_{j=1}^{N_l} \{\exp[-||\mathbf{f}_{lj} - \mathbf{f}^*||^2_{\mathbf{w}^*}]\} \tag{6.2}$$

This leads to the max-ed version of Eqn. (6.1) on positive bags:

$$Q_l = \begin{cases} e^{-||\mathbf{f}_{lJ(u_l)} - \mathbf{f}^*||^2_{\mathbf{w}^*}} & , \ y_{I_l} = 1 \\ \prod_{j=1}^{N_l} (1 - e^{-||\mathbf{f}_{lj} - \mathbf{f}^*||^2_{\mathbf{w}^*}}) & , \ y_{I_l} = -1 \end{cases} \tag{6.3}$$

The DD function in Eqn. (6.3) is used to construct an objective function Q over all bags, $Q = \prod_{l=1}^{L} Q_l$. Q is maximized by an EM algorithm [29].

We use each instance in each positive bag to repeatedly initiate the DD-optimization process presented above, and prototypes with DD values smaller than a threshold H_{dd} (that equals to the mean of DD values of all learned prototypes) are excluded. Such a prototype learning process is conducted for each semantic concept independently, and the final learned prototypes construct a codebook to describe the discriminative multi-modal characteristics of each individual concept.

In practice, since the number of negative bags is usually much larger than that of positive bags, we maintain a balanced number of positive and negative bags for prototype learning by sampling the negative ones. Specifically, the negative bags that come from the same videos as positive bags are all used, and at least one negative bag is randomly selected from the remaining videos.

6.6 Classification with Joint A-V Codebooks

For each semantic concept C^m, the learned prototypes form a codebook to describe its discriminative characteristics, each prototype corresponding to a codeword. These codewords span a codebook-based feature space to represent AVAs. For an AVA with a long-term region track $\mathbf{r}_j$ and a feature $\mathbf{f}_j$, it can be mapped to each prototype codeword $(\mathbf{f}_p^{m*}$,$\mathbf{w}_p^{m*})$ by the weighted norm-2 distance $||\mathbf{f}_j - \mathbf{f}_p^{m*}||^2_{\mathbf{w}_p^{m*}}$. Accordingly, each video segment u can be mapped to each prototype codeword by using the minimum distance $D(u,\mathbf{f}_p^{m*})_{\mathbf{w}_p^{m*}} = \min_{\mathbf{r}_j \in u}\Big\{||\mathbf{f}_j - \mathbf{f}_p^{m*}||^2_{\mathbf{w}_p^{m*}}\Big\}$. Then the video segment u can be represented by a codebook-based feature $\mathbf{D}^m(u) = \Big[D(u,\mathbf{f}_1^{m*})_{\mathbf{w}_1^{m*}},\ldots,D(u,\mathbf{f}_{P_m}^{m}{}^*)_{\mathbf{w}_{P_m}^{m}{}^*}\Big]^T$, base on which classifiers like SVMs [230] can be trained for concept detection.

By using different combinations of $\mathbf{f}_{vis}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{loc}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$, various codebooks can be generated in different multi-modal feature spaces. In general, different types of codebooks have uneven advantages at detecting different concepts. We selectively choose the optimal types of codebooks to use by adopting a boosting feature selection framework similar to [219]. The Real AdaBoost method [59] is used where during each iteration, an optimal codebook is selected to construct an SVM classifier as the weak learner, and the final detector is generated by adding up weak learners from multiple iterations. The boosting algorithm is summarized in Fig. 6.14.

6.7 Experiments

We evaluate our algorithm over Kodak's consumer benchmark video set [137]. As described in Chapter 2, Section 2.7.1, the data set contains 1358 videos from real consumers. 5166 keyframes are uniformly sampled from the videos for every 10 seconds, and are labeled to

Input: Training set $\mathcal{D} = \{(u_1, y_{I_1}), \dots, (u_L, y_{I_L})\}$. Each video segment u_l is represented by several types of codebook-based features learned with different combinations of $\mathbf{f}_{vis}$, $\mathbf{f}_{mt}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{loc}$, and $\mathbf{f}_{audio}$.

1. Initialization: set sample weights $\sigma_l^1 = 1/2L^+$ or $1/2L^-$ for $y_{I_l} = 1$ or -1, respectively, where L^+ (L^-) is the number of positive (negative) samples; set final decisions $H^1(u_l) = 0$, $l = 1, \dots, L$.

2. Iteration: for $\tau = 1, \dots, \Gamma$

- Get training set $\tilde{\mathcal{D}}^\tau$ by sampling $\mathcal{D}$ according to weights σ_l^τ.
- Train an SVM over set $\tilde{\mathcal{D}}^\tau$ by using the k-th type of feature. Get the corresponding $q_k^\tau(u_l) = p_k^\tau(y_{I_l} = 1|u_l)$, $l = 1, \dots, L$.
- Set $h_k^\tau(u_l) = \frac{1}{2} \log \frac{q_k^\tau(u_l)}{1 - q_k^\tau(u_l)}$.
- Choose the optimal $h^{\tau,*}(\cdot) = h_k^\tau(\cdot)$ with the minimum error $\epsilon^{\tau,*} = \epsilon_k^\tau$, $\epsilon_k^\tau = \sum_{l=1}^N \sigma_l^\tau e^{-y_{I_l}(H^\tau(u_l) + h_k^\tau(u_l))}$, $\epsilon_k^\tau < \epsilon_j^\tau$ if $j \neq k$.
- Update weights: $\sigma_l^{\tau+1} = \sigma_l^\tau e^{-y_{I_l} h^{\tau,*}(u_l)}$, $l = 1, \dots, L$, and re-normalize so that $\sum_{l=1}^L \sigma_l^{\tau+1} = 1$.
- Update $H^{\tau+1}(u_l) = H^\tau(u_l) + h^{\tau,*}(u_l)$ for $l = 1, \dots, L$.

Figure 6.14: The algorithm to construct concept detectors by selectively using different codebooks. 10 iterations are empirically taken in our experiments ($\Gamma = 10$).

21 semantic concepts that are of great interest based on real user study [1]. We separate the entire data set into two subsets: 60% videos, *i.e.*, 813 videos, are randomly sampled as the training data; and the rest 40% videos are used for testing. One-vs.-all classifiers are trained for detecting each individual concept. The AP and MAP are used as performance evaluation metrics.

To extensively evaluate the proposed audio-visual analysis framework, we experiment on two different concept detectors using the audio-visual atomic representation: (1) *Visual Atoms with MIL codebook construction* (*VA-MIL*), where the visual-codebook-based features are directly used to train SVM detectors; and (2) *AVA with MIL codebook construction and Boosting feature selection* (*AVA-MIL-Boosting*), where different types of codebooks are generated and selectively used via Boosting. In addition, we compare VA-MIL with two

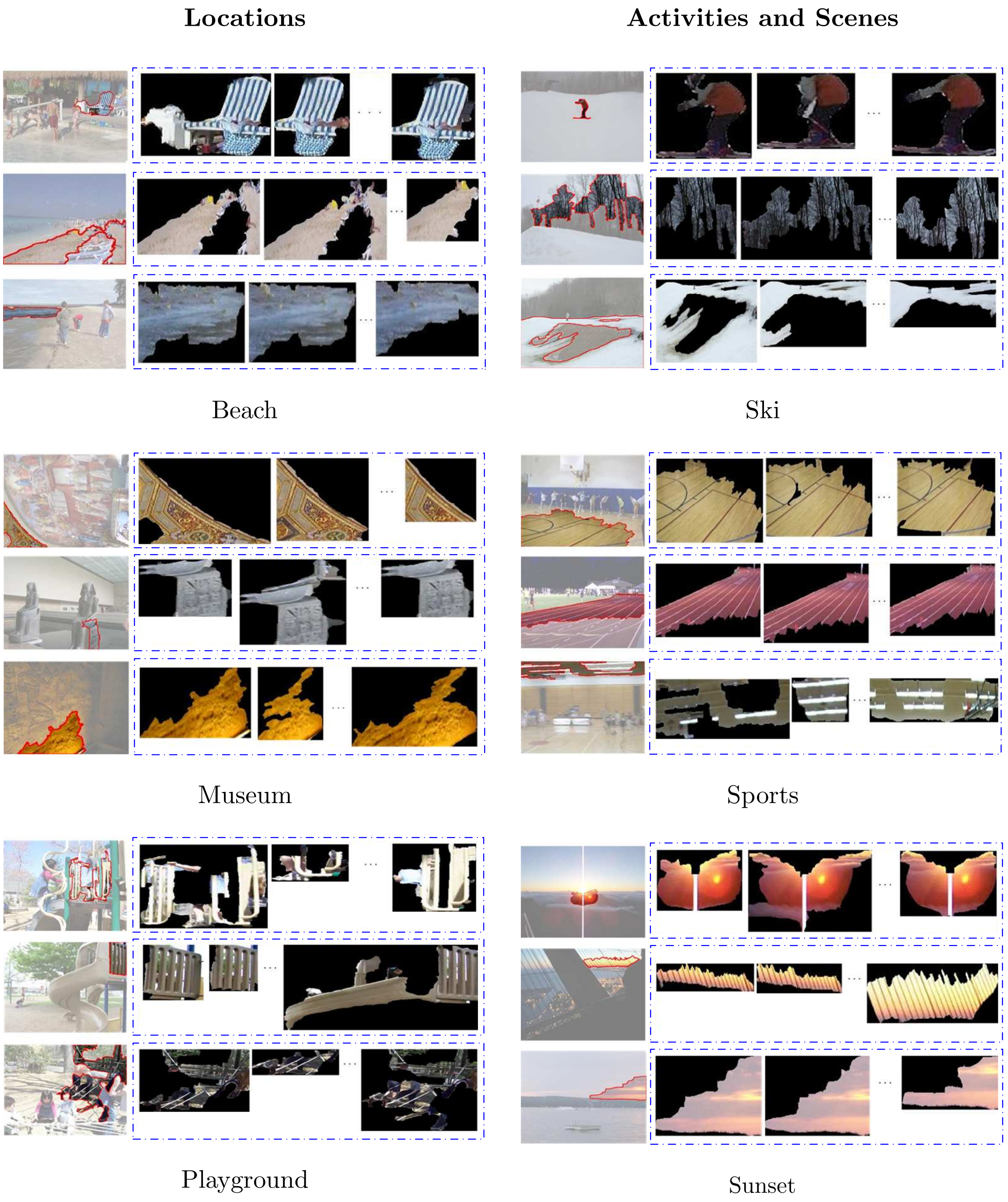
[1] The same 21 semantic concepts used in the previous experiments in Chapter 5

state-of-the-art static-region-based image categorization approaches that also use MIL, *i.e.*, DD-SVM [29] and ASVM-MIL [251]. For static-region-based methods, each video-segment bag u contains a set of static regions that come from the center frame of each short-term video slice v in this bag. DD-SVM learns visual codebooks with static bags using MIL for individual concepts, and codebook-based features are generated to train SVMs. ASVM-MIL directly builds an asymmetrical SVM over the static regions under the MIL setting. No temporal tracking is involved in both of these two approaches.

6.7.1 Codebook Visualization

We first show examples of the discriminative prototypes learned in various types of codebooks. Such visualization helps us to subjectively and intuitively evaluate different approaches. To get the AVAs to visualize, for each prototype $(\mathbf{f}^*, \mathbf{w}^*)$ we calculate the distances between all training AVA instances and this prototype. Then the AVA with the minimum distance is considered as the most appropriate example to visualize this prototype. In addition, prototypes learned for each concept can be ranked according to the DD values Q in descending order. The higher rank a prototype has, the better the prototype describes the discriminative characteristics of this concept. Fig. 6.15 gives some example prototypes (ranked within top 50) extracted based on audio-visual atoms. From Fig. 6.15, the audio-visual-atom-based prototypes are very reasonable. For example, in the Location category we get water, sand, and beach facility as representative patterns to describe the "beach" concept; in the Activity category, we get the white snow, bald white trees, and the athlete as representative patterns to describe the "ski" concept; in the Occasion category, we get wedding gown, black suit, and wedding candles as representative patterns to describe the "wedding" concept; and in the Object category we get the baby face, baby hand, and baby toys as representative patterns to describe the "baby" concept.

In comparison, Fig. 6.16 gives some example prototypes learned by the static-region-based DD-SVM algorithm. In later experiments we will see that our method significantly outperforms static-region-based approaches. The prototype visualization helps to explain such results. The static DD-SVM gets very noisy prototypes in general, *e.g.*, many fragments



Locations

Activities and Scenes

Beach

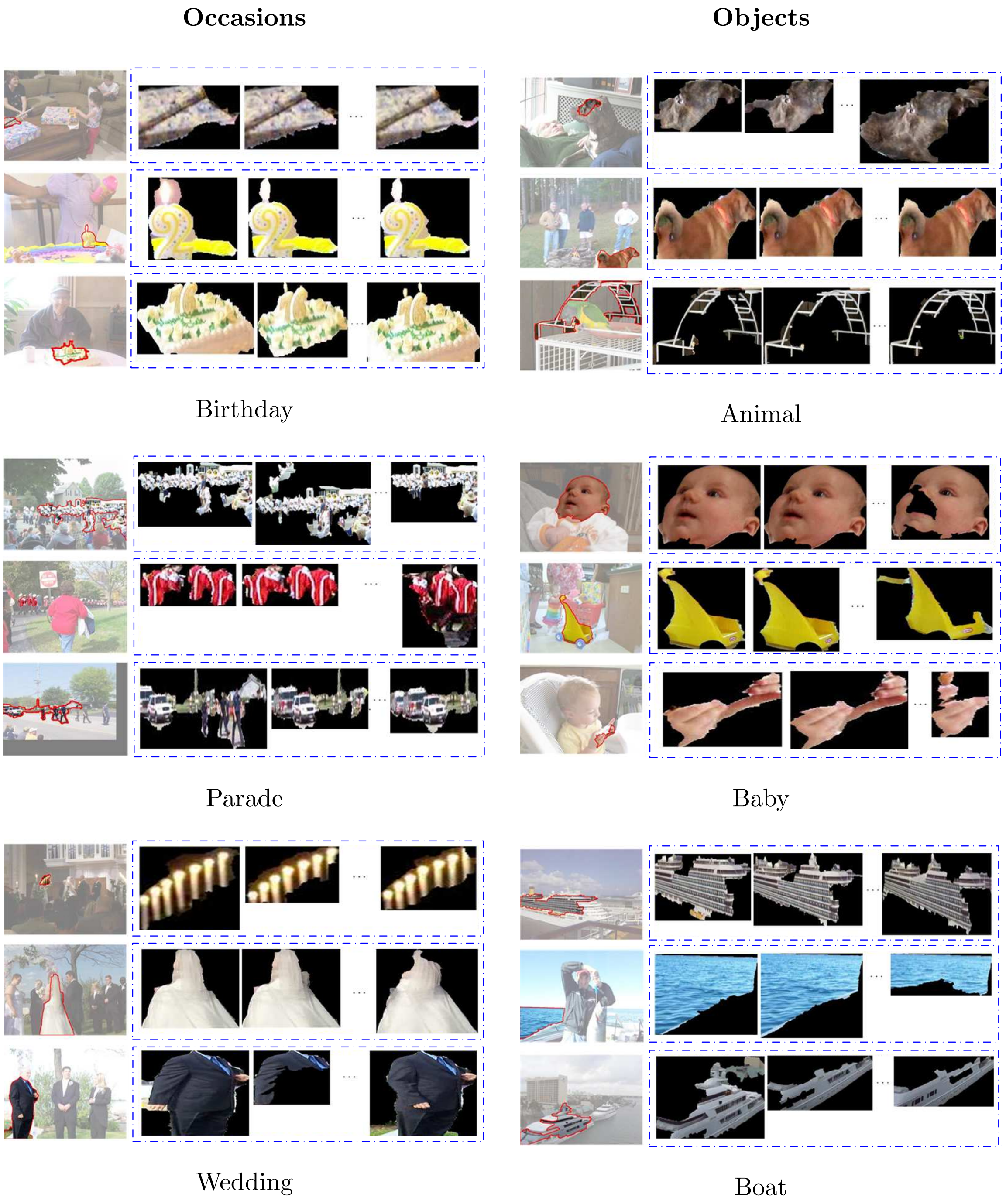
Ski

Museum

Sports

Playground

Sunset



Occasions

Objects

Birthday

Animal

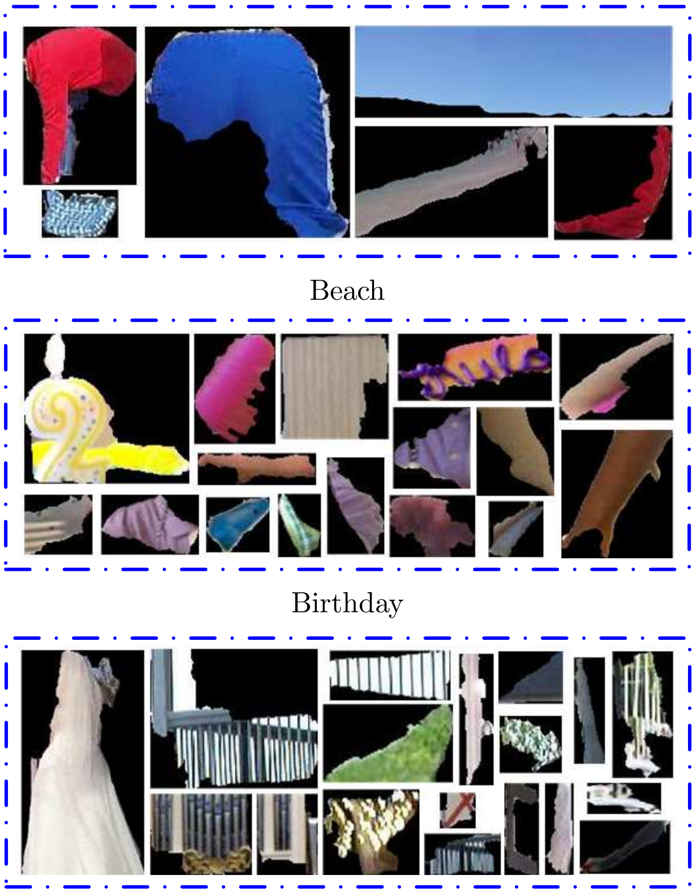
Parade

Baby

Wedding

Boat

Figure 6.15: Example prototypes learned with audio-visual atoms for concepts of several broad categories, *i.e.*, Locations, Activities, Scenes, Occasions, and Objects. Doted blue boxes show the corresponding region track prototypes, where images on the left show example frames where region tracks are extracted.



Beach

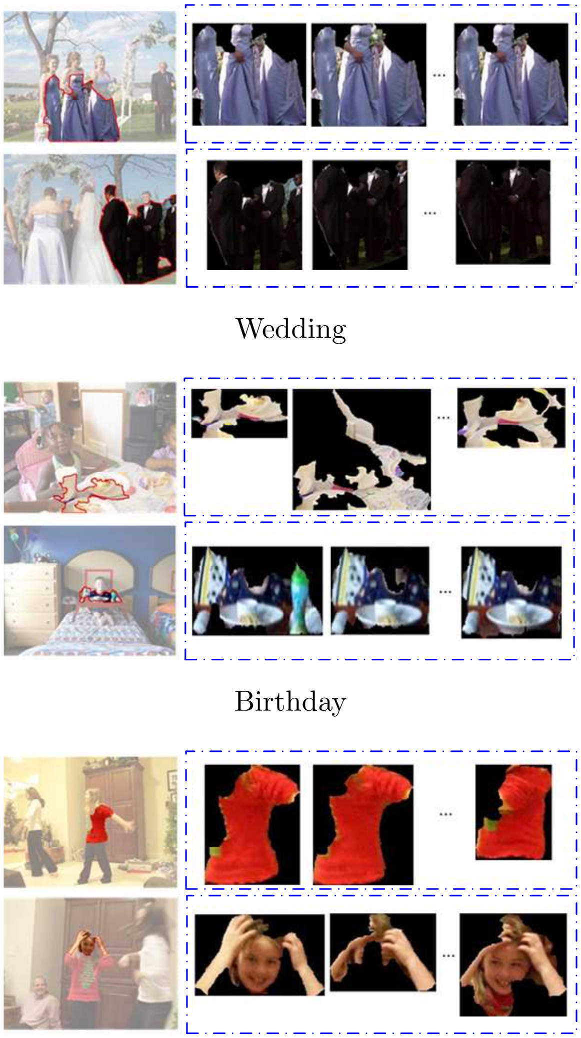
Birthday

Wedding

Figure 6.16: Examples of learned static region prototypes by DD-SVM. The static codebook contains lots of noise, *e.g.*, fragments of human clothes, which causes severe degradation of the concept detection performance.

of human clothes are extracted as representative patterns. Although some good prototypes can also be obtained, the performance suffers from the noisy ones a lot. The results also confirm our motivation that the region tracks are more noise-resistent for video concept detection than static regions.

By adding audio features to visual atoms, salient audio-visual patterns can be discovered by the audio-visual codebook for concepts that are expected to have strong cues in both audio and visual aspects. Fig. 6.17 gives some example prototypes learned by using AVAs with the concatenation of $\mathbf{f}_{vis}$ and $\mathbf{f}_{audio}$. These prototypes are salient for concept detection,



Wedding

Birthday

Dancing

Figure 6.17: Example prototypes learned with audio-visual atoms for concepts that are expected to have strong cues in both audio and visual aspects. These prototypes are not discovered by visual-only codebooks. For example, the salient patterns about the tableware with birthday cake inside can be found by considering audio and visual features jointly but are not learned by using visual features alone. This is because tableware also appears in many other videos visually, and only when combined with the background birthday music can the tableware generate a salient audio-visual pattern for "birthday" concept.

but are not captured by visual-atom-based codebooks. For example, those salient patterns about the tableware with a piece of birthday cake inside can be discovered by considering audio and visual features jointly but can not be extracted by using visual features alone. This is because tableware also appears in many other videos visually, and only when combined with the background birthday music can the tableware generate salient audio-visual cues to describe "birthday" videos. Similarly, body parts of a dancing person can be discovered by using audio-visual atoms but are missed by using visual features alone, since only when combined with background music can the body parts form salient audio-visual cues to describe "dancing" videos.

6.7.2 Performance of Concept Detection

In this section, we compare AP and MAP of different algorithms for semantic concept detection. All algorithms use the RBF kernel $k(\mathbf{x}_1, \mathbf{x}_2) = \exp\{-\gamma||\mathbf{x}_1 - \mathbf{x}_2||^2\}$, and the multiple-parameter technique [24]. That is, the error control parameter C in SVM [230] takes values $C{=}2^s$, $s{=}\{0, 1, 2, 3, 4\}$, and γ takes values $\gamma = (1/d)^{2^t}$, $t = \{-3, -2, -1, 0, 1\}$ (d is the dimension of the data point $\mathbf{x}$). This gives 25 parameter settings with different combinations of C and γ, based on which 25 SVM classifiers can be constructed and then averagely fused to generate the final classification result. We use this multi-parameter technique instead of tuning parameters by cross-validation due to the findings in [24], *i.e.*, over this challenging consumer video set, parameter tuning tends to over fit resulting from the large diversity of the video content.

6.7.2.1 Region Tracks vs. Static Regions

As described in Section 6.6, each visual atom is associated with a visual feature vector $\mathbf{f}_{vis}$. For a static region, we can also extract a visual feature $\mathbf{f}_{vis}$. Fig. 6.18 shows the per-concept AP and MAP comparison of VA-MIL, DD-SVM, and ASVM-MIL, by using $\mathbf{f}_{vis}$. The results from random guess are also shown for comparison. From the figure, our VA-MIL consistently outperforms other methods over every concept, and significant performance improvements, *i.e.*, over 120% MAP gain on a relative basis, can be achieved compared to both DD-SVM and ASVM-MIL. This phenomenon confirms that static regions segmented from general videos are very noisy. Our visual atoms can significantly reduce the noise by not only extracting robust and trackable regions, but also averaging out the outlier noise through the entire tracked sequence.

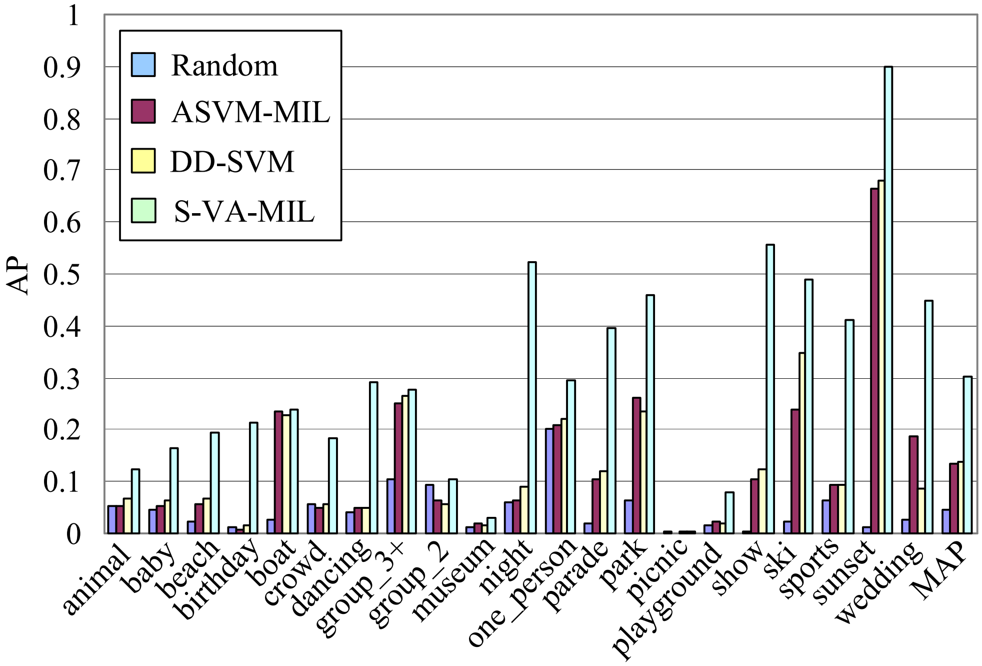


Figure 6.18: Comparison of visual atoms with static-region-based methods.

6.7.2.2 AVA-MIL with Multi-Modal Features

Fig. 6.19 gives the performance comparison of our AVA-MIL by using different codebooks generated from individual $\mathbf{f}_{vis}$, $\mathbf{f}_{mt}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{loc}$, and $\mathbf{f}_{audio}$. From the result, $\mathbf{f}_{mt}$ performs badly because of the low-quality motion in general videos and the lack of discriminative power of motion alone for concept detection. For example, the moving speed and direction of a person can not discriminate "one person" videos. Also, $\mathbf{f}_{loc}$ does not perform well due to the lack of discriminative power of spatial location alone for concept detection. In general, audio feature $\mathbf{f}_{audio}$ alone can not compete with visual feature $\mathbf{f}_{vis}$ or $\mathbf{f}_{sift}$, since most of the 21 concepts are visual-oriented. However, $\mathbf{f}_{audio}$ works very well over "museum". The visual content of "museum" videos is very diverse while the audio sound is relatively consistent, *e.g.,* the sound of people talking and walking in a large quiet indoor room. The global $\mathbf{f}_{vis}$ outperforms local $\mathbf{f}_{sift}$ over most of the concepts except for "dancing" and "sports". As discussed in detail in Chapter 2 Section 2.7.5, this is because of the challenging condition to detect generic concepts in this Kodak's video collection. Due to the large diversity in the visual content, there is very little repetition of objects or scenes in different videos, even those from the same concept class. In such a case, it is hard to exert the advantage of local descriptors like SIFT for local point matching/registration.

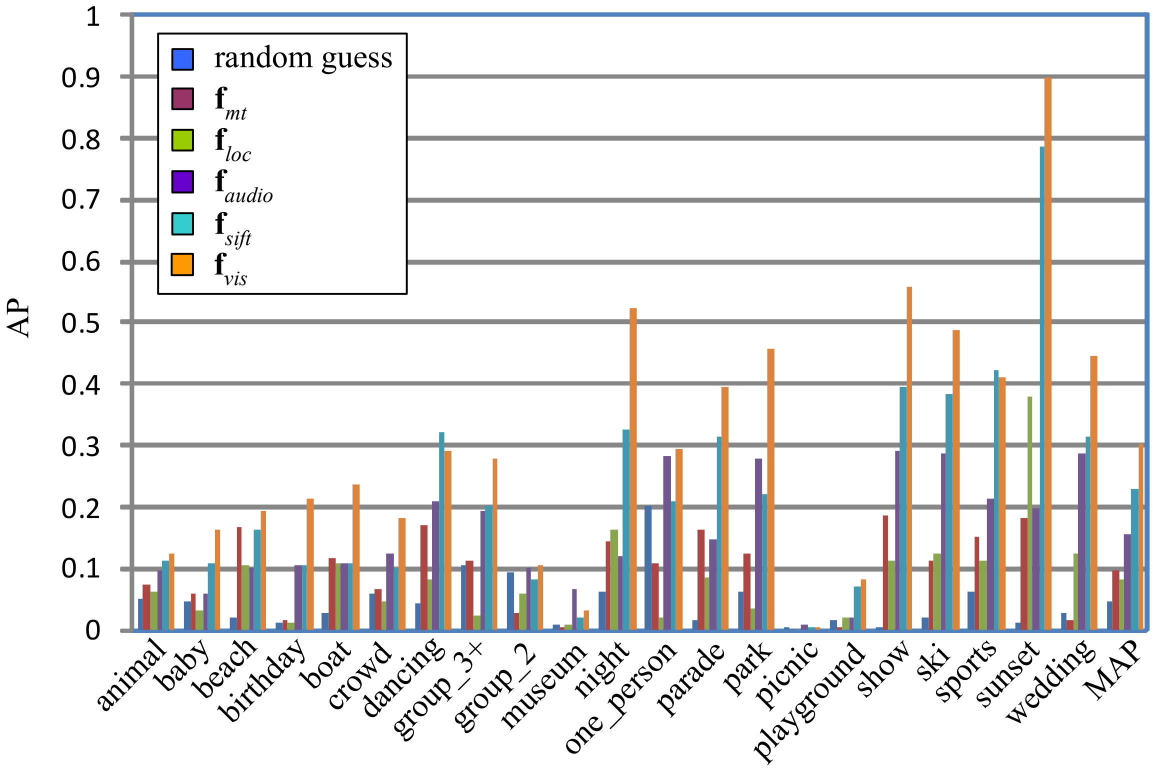


Figure 6.19: Comparison of AVA-MIL with individual $\mathbf{f}_{vis}$, $\mathbf{f}_{loc}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$.

Fig. 6.20 gives the AP and MAP comparison of AVA-MIL using different combinations of $\mathbf{f}_{vis}$, $\mathbf{f}_{loc}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$. From the result, compared with individual $\mathbf{f}_{vis}$, both of the multi-modal features generated by concatenating $\mathbf{f}_{vis}$ and $\mathbf{f}_{mt}$, and $\mathbf{f}_{vis}$ and $\mathbf{f}_{audio}$ have slight improvements in terms of MAP. By adding the noisy motion features ($\mathbf{f}_{vis}$+$\mathbf{f}_{mt}$), most concepts get worse or unchanged performances except for "beach" and "sports", which receive 9.6% and 3.5% AP gains, respectively. This is reasonable since "sports" videos often have fast moving athletes, and "beach" videos have large stable regions like sky, sand and waterfront that do not move. By adding spatial features ($\mathbf{f}_{vis}$+$\mathbf{f}_{loc}$), most concepts also get performance degradation, due to the lack of discriminative power of the spatial location information in detecting the set of semantic concepts we experiment on. By adding SIFT features ($\mathbf{f}_{vis}$+$\mathbf{f}_{sift}$) we can not get overall improvement either. By adding audio features ($\mathbf{f}_{vis}$+$\mathbf{f}_{audio}$), 12 concepts get clear improvements, *e.g.,* "boat" and "sports" get 28.7% and 15.9% AP gains, respectively. However, we also have noticeable AP degradation over some concepts like "crowd" and "park", because the regional color and texture features are much more powerful in detecting these concepts than audio features. The results also indicate the uneven strengths of different modalities in detecting different concepts. Therefore, as will be shown in Section 6.7.2.3, a more rigorous approach in selecting optimal features from different modalities is needed.

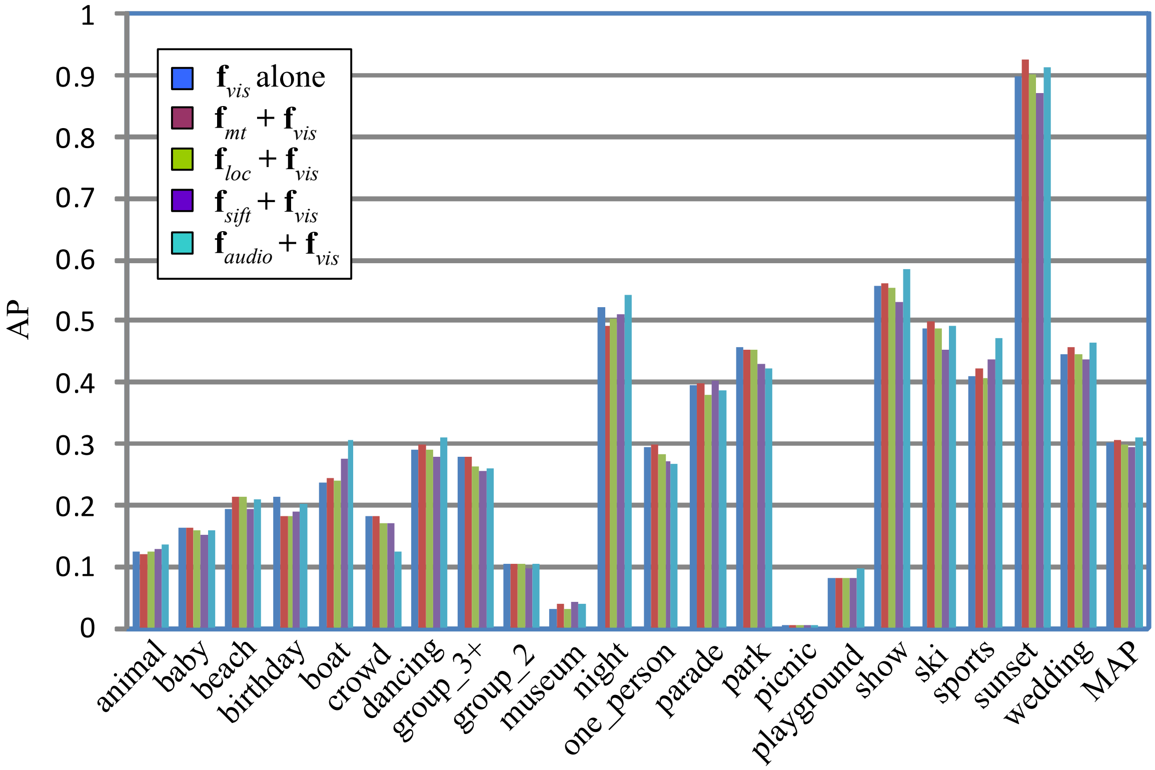


Figure 6.20: Comparison of AVA-MIL with different combinations of $\mathbf{f}_{vis}$, $\mathbf{f}_{loc}$, $\mathbf{f}_{sift}$, $\mathbf{f}_{mt}$, and $\mathbf{f}_{audio}$.

6.7.2.3 AVA-MIL-Boosting with Multi-Modal Features

Fig. 6.21 gives the performance of multi-modal AVA-MIL-Boosting. For better comparison, we show results from VA-MIL using visual feature $\mathbf{f}_{vis}$ alone. Also, we compare with a straightforward fusion approach, where SVMs trained using codebooks generated from $\mathbf{f}_{vis}$, $\mathbf{f}_{vis}+\mathbf{f}_{mt}$, and $\mathbf{f}_{vis}+\mathbf{f}_{audio}$, respectively, are averagely combined together to give the final detection results. From the figure, we can see that by selectively using the optimal types of codebooks for detecting different individual concepts, the multi-modal AVA-MIL-Boosting can improve the detection performance over most of the concepts (17 out of 21) compared to VA-MIL. Significant AP gains are achieved (on a relative basis) for “animal” by 20.9%, “beach” by 28.3%, “boat” by 35.4%, “crowd” by 11.7%, “group of three or more” by 14.6%, “dancing” by 56.7%, “museum” by 106.3%, “playground” by 20.7%, and “sports” by 18.9%. The overall MAP is improved by 8.5%. In comparison, without feature selection, the direct fusion method can not improve the overall performance by simply adding up classifiers from different modalities, which is consistent with the findings in [24].

6.8 Discussion

We propose a framework for audio-visual analysis in general videos by extracting atomic representations over video segments. Visual atoms are extracted by a hierarchical framework where in the first step an STR-PTRS algorithm is developed to track consistent visual regions within short-term video slices and then in the second step the extracted short-term region tracks are connected into visual atoms by visual linking. At the same time, the corresponding audio soundtrack is decomposed to time-frequency bases, according to which audio energy onsets are located. Audio atoms are reconstructed from the time-frequency bases around each audio energy onset. Visual atoms are associated with audio atoms to generate AVAs. A set of regional visual features and a spectrogram audio feature are assigned to each AVA. Joint audio-visual codebooks are constructed on top of the AVAs to capture salient audio-visual patterns for effective concept detection. Our method provides a balanced choice for audio-visual analysis in general videos: we generate a middle-level atomic representation to fuse visual and audio signals and do not rely on precise object extraction.

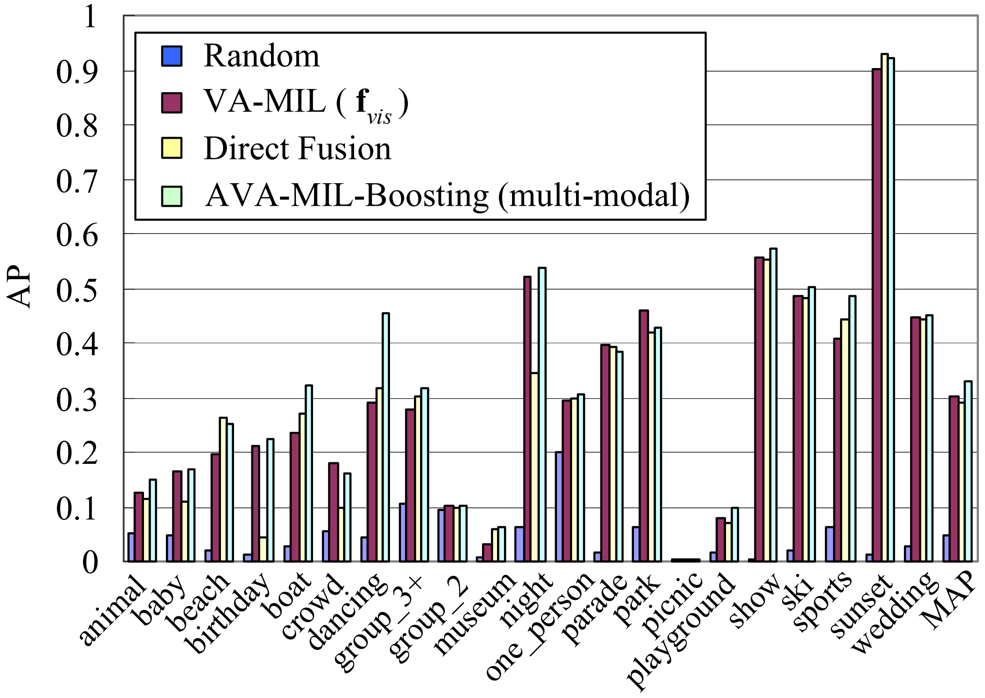


Figure 6.21: Comparison of multi-modal AVA-MIL-Boosting, VA-MIL with $\mathbf{f}_{vis}$ alone, and direct fusion.

Experiments over the challenging Kodak's consumer benchmark videos demonstrate the effectiveness.

One major future work is to explore moderate-level audio-visual synchronization, *i.e.*, sounding regions, from general videos. As discussed in Section 6.3, with a backward checking process, our STR-PTRS can be extended to find short-term region tracks with their starting/ending time stamps within video slices. Trajectories of such short-term region tracks can be generated. On the other hand, the MP-based audio representation is specifically chosen to be able to find audio features corresponding to identified video objects. By describing audio as a set of parameterized basis functions, distinct features of overlapping sounds can be segmented into disjoint sets of functions. Since each function depends only on a compact neighborhood of time-frequency energy, it is largely invariant to simultaneous energy from other sources, unlike common audio features like MFCCs which reflect the global properties of all energy present in a sound. Moreover, the precise timing associated with each function can provide both for the detection of repeated structure within the sound, and also for detailed synchronous correlation against video features. By modeling relations between visual region trajectories and audio bases, we can study moderate-level audio-visual synchronization for discovering interesting audio-visual events like horse running and people singing. In addition, we may explore temporal patterns beyond co-occurrence and synchronization, such as “audio atom A typically precedes video atom B by a time offset between 1-3 seconds”. We may investigate related data mining techniques to discover such pattern rules.

Chapter 7

Conclusions

In this final chapter, we will conclude the work presented in this thesis along the two directions that we have explored to enhance generic concept detection in general videos. We will then present a few potential areas for extension and improvements.

7.1 Summary

This thesis is dedicated to the research in detecting generic concepts from general videos focusing on two main research areas: leveraging information from multiple modalities and exploring inter-conceptual relationships.

7.1.1 Multi-Concept Learning

We have developed two different multi-concept learning approaches. The first method, BCRF-CF, models the inter-conceptual relationships by a CRF that takes as input the posteriors predicted from individual concept detectors and outputs refined detection results through graph inference. To avoid the difficulty of designing two-node compatibility potentials in CRF, a discriminative objective function aiming at class separation is optimized by iterative boosting. In addition, we have proposed a simple but effective criterion to predict which concepts will benefit from CBCF, based on both the strength of relationships between a concept and its neighborhood and the robustness of detectors of this neighborhood.

Based on the framework of BCRF-CF we have also explored the interesting "20 questions problem" for concept detection, where the user's interaction is incorporated. We have proposed an Active CBCF paradigm, where for each input data, the system actively selects a small number of concepts for the user to label and then uses user's annotation to help determine the presence of other concepts, by utilizing the inter-conceptual relationships. We have presented an accurate method to choose the most informative concepts given the specific input image, for the user to label based on both information theoretic and heuristic rules.

Furthermore, instead of modeling inter-conceptual relationships explicitly in the two-layer CBCF framework, we have proposed the second multi-concept learning method to construct a single-layer multi-class concept detector by sharing kernels and classifiers through an iterative joint boosting framework. Multiple kernels over multi-resolution visual vocabularies and multi-resolution spatial coordinates are constructed and shared among different concepts to improve the accuracy of detecting individual concepts.

We have evaluated our multi-concept learning algorithms over the TRECVID 2005 news video set. Promising results have been obtained in terms of both the improved detection accuracy and efficiency. Our prediction method also allows us to apply multi-concept learning when it is likely to be effective.

7.1.2 Multi-Modality Learning

To incorporate information from both visual and audio signals, we have developed algorithms exploring both multi-modality fusion and mid-level audio-visual correlation. Two multi-modality fusion methods have been presented: the early fusion A-V Joint Boosting and the late fusion A-V Boosted CRF. A-V Joint Boosting extends the kernel/classifier sharing algorithm developed for multi-concept learning, which constructs and shares both visual-based and audio-based kernels and classifiers to build multi-class concept detectors. A-V Boosted CRF extends the BCRF-CF algorithm, where the input of CRF includes both audio-based and visual-based independent concept detection results. Both A-V Joint Boosting and A-V Boosted CRF provide natural combinations of multi-concept learning and multi-modality learning, and are flexible to incorporate other types of modalities, *e.g.*, textual if available, to further help classification.

To conduct mid-level audio-visual correlation in general videos, we have proposed to extract atomic representations over video segments, which does not rely on precise object extraction or tight audio-visual synchronization. Specifically, a hierarchical framework has been developed to get visual atoms from the visual channel. In the first step consistent visual regions are tracked within short-term video slices, and then in the second step such short-term region tracks are connected into visual atoms based on visual similarities. At the same time, the corresponding audio soundtrack is decomposed to time-frequency bases, according to which audio energy onsets are located. Audio atoms are reconstructed from the time-frequency bases around each audio energy onset. Visual atoms are associated with audio atoms to generate audio-visual atoms, and a set of visual features and audio features are assigned to such atoms. Joint audio-visual codebooks are constructed on top of the atoms to capture salient audio-visual patterns for concept detection.

We have evaluated our multi-modality learning approaches over Kodak's consumer benchmark video set. Experimental results demonstrate that both audio and visual features contribute significantly to the robust detection performance, and that by combining information from both audio and visual channels we can achieve better detection than using single modalities alone.

7.2 Future Work

Generic concept detection in general videos is a challenging problem in general, and many research efforts have been made to improve the detection robustness, accuracy, and speed. Among various approaches, multi-modality and multi-concept learning continues to attract interest from many researchers. These two directions are not mutually exclusive, and we can combine them to exploit the power of both. We will discuss some interesting directions below, including those for improving multi-concept learning or multi-modality learning alone, and those aiming at more effective fusion.

7.2.1 Multi-Concept Learning

Most of the current multi-concept learning works use concept co-occurring statistics in images or videos to generate inter-conceptual relationships. Such approximations can be unreliable when we only have limited training data, which, unfortunately, is usually the case in practice due to the high cost in obtaining ground-truth sets. On the other hand, with the proliferation of image and video sharing websites such as Flickr and YouTube, conceptual structures and relationships can be obtained through mining the textual information associated with web images and videos. The rich online multimedia resources give promises to accurate estimation of the complicated relationships over a large number of real-world semantic concepts. In such cases, multi-concept learning can be combined with techniques such as Web data mining to build large-scale concept detection systems where a huge number of concepts as well as their relationships can be studied. One interesting work along this direction was described in [102]. More research may need to be done to adapt and apply concept relations obtained from online multimedia data to specific individual collections, *e.g.*, consumers' personal albums.

7.2.2 Multi-Modality Fusion

As mentioned earlier, the multi-modality fusion algorithms proposed in this thesis are flexible to incorporate other types of modalities in the future. In A-V Joint Boosting kernels/features from other modalities, *e.g.*, textual description or other meta data, can be combined with visual and audio kernels/features to enhance classification. In A-V Boosted CRF, individual detection results from independent concept detectors trained over other modalities can also be combined with audio-based and visual-based independent detection results to feed into the inter-conceptual CRF.

The general issues that remain unsolved related to multi-modality fusion include: how to normalize and combine features extracted from different signals [1]; how to normalize and combine judgements generated by classifiers learned over features of different modalities;

[1] The goal of feature normalization is to ensure that the contribution of each feature component to the learned model is comparable.

how to conduct dimension reduction in the high-dimension multi-modal feature space; how to select optimal features/classifiers for each specific task.

7.2.3 Mid-Level Audio-Visual Correlation

7.2.3.1 Improving Audio-Visual Atoms

The quality of the current AVAs is limited by the quality of image segmentation. Although by temporal tracking we alleviate the problem of noisy segments, bad segments (caused by sudden movement of camera or sudden lighting change) can still break temporal tracking. To increase the robustness of the extracted AVAs, several approaches can be taken, such as using multiple segmentation strategies to generate various sets of segments, or using overlapping video slices with multiple window durations to generate a large pool of candidate region tracks.

In addition, as discussed in Section 6.3, the current AVAs can not capture objects than enter and leave the screen within a short video slice. We can extend the STR-PTRS algorithm with a backward tracking process or bi-directional tracking process, so that we can not only capture such objects but also locate their starting/ending time stamps within the video slice.

The current audio atoms and visual atoms are extracted from video segments in an unsupervised manner, which may not correspond to semantic meanings. In the future, supervised/semi-supervised learning approaches can be introduced to extract semantically more meaningful audio and visual atoms.

7.2.3.2 Audio-Visual Atomic Representation

The joint atoms offer a natural representation to describe videos in terms of a number of middle-level features that are more relevant to human observers than low-level features. It captures essential characteristics of videos, such as the presence of regions (ideally objects) with consistent distinct appearance, sound, and motion. A systematic study can be conducted in the future to use such joint representation for a broad range of applications such as audio-visual pattern mining, audio-visual object/event detection and localization, and retrieval and summarization of general videos. We elaborate on these ideas below.

To describe the content of large multimedia collections, representative audio-visual patterns, in the form of AVA templates, can be constructed on top of the AVAs extracted from individual videos. Such audio-visual patterns can be discovered in a supervised process (*e.g.*, learning of discriminative patterns to distinguish a specific personal collection), or an unsupervised or semi-supervised process (*e.g.*, discovery of frequently occurring patterns in a consumer set). For instance, through mining of a general video collection related to wedding events, we can find the salient audio-visual patterns consistently occurring in wedding videos. By examining the difference between wedding videos from a specific personal collection and those from general sources (*e.g.*, wedding videos from the internet), unique audio-visual patterns can be found to distinguish this individual personal collection. Such mining results are helpful to organize, summarize, and browse multimedia data.

Complicated semantic events, *e.g.*, wedding, birthday, and graduation, usually consist of some salient audio-visual elements. For example, the birthday event is usually associated with birthday cake, birthday candle, birthday balloon, birthday music, birthday card, *etc.* The AVAs or the higher-level audio-visual patterns not only capture important audio-visual characteristics of these elements, but also provide the ability to study correlations between these elements. For instance, the presence of a cake does not necessarily indicate a birthday event, but the co-occurrence of a cake with a candle and a balloon significantly increases the confidence for birthday detection. Constructing event models by taking into account both the individual AVAs and their co-occurring, spatial, or temporal relationships is an interesting direction for future research.

A natural application following the audio-visual pattern mining and object/event detection is content-based retrieval, *e.g.*, retrieval of items whose content, action, or location (based on both audio and visual features) resemble a query example or small set of examples. Interesting research topics include new relevance feedback strategies, new similarity measurements, and new classification frameworks.

7.2.4 Fusion of Multi-Concept & Multi-Modality Learning

As discussed before, the A-V Joint Boosting and A-V Boosted CRF algorithms presented in this thesis combines multi-modality learning and multi-concept learning naturally. Since real-world concepts are interrelated and are innately multi-modal, it is intuitive to design concept detection systems that consider both multi-concept and multi-modality learning jointly. For instance, users usually organize multimedia data according to highly semantic event occurrences, such as Vivian's birthday party. Such events are generally composed by many concept elements, including both visual-oriented concepts like people (*e.g.*, Vivian and her friends) and birthday cake, and audio-oriented concepts like music and birthday song. By visual, audio, and joint audio-visual analysis as well as studying the relationships of interrelated multi-modal concepts, we can build systematic models to describe and classify these events.

Bibliography

[1] M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.

[2] A. Amir, J.O. Argillander, M. Berg, S.F. Chang, M. Franz, W. Hsu, G. Iyengar, J.R. Kender, L. Kennedy, C.Y. Lin, M. Naphade A. Natsev, J.R. Smith, J. Tesic, G. Wu, R. Yan, and D. Zhang. Ibm research trecvid-2004 video retrieval system. *Proc. NIST TRECVID Workshop*, 2004.

[3] R. Arrighi, F. Marini, and D. Burr. Meaningful auditory information enhances perception of visual biological motion. *Journal of Vision*, 9(4):1–7, 2009.

[4] J.J. Aucouturier and F. Pachet. Music similarity measures: What's the use? *Proc. International Conference on Music Information Retrieval*, 2002. Paris, France.

[5] F.R. Bach, G.R.G. Lanckriet, and M.I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. *Proc. IEEE International Conference on Machine Learning*, pages 41–48, 2004.

[6] D. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.

[7] Z. Barzelay and Y. Schechner. Harmony in motion. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. Minneapolis, MN.

[8] B. Bascle and R. Deriche. Region tracking through image sequences. *Proc. International Conference on Computer Vision*, pages 302–307, 1995.

[9] A. Baumberg. Reliable feature matching across widely separated views. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 774–781, 2000.

[10] M.J. Beal, N. Jojic, and H. Attias. A graphical model for audiovisual object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):828–836, 2003.

[11] P.R. Beaudet. Rotationally invariant image operators. *Proc. International Joint Conference on Pattern Recognition*, pages 579–583, 1978.

[12] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4), 2002.

[13] S. Birchfield. Klt: An implementation of the kanade-lucas-tomasi feature tracker. 2007. http://vision.stanford.edu/∼birch.

[14] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *Proc. IEEE International Conference on Computer Vision*, 2:1395–1402, 2005.

[15] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267, 2001.

[16] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[17] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. *Proc. ACM International Conference on Image and Video Retrieval*, pages 401–408, 2007.

[18] A. Bosch, A. Zisserman, and X. Munoz. Scene classification using a hybrid generative/dicriminative approach. *IEEE Trasactions on Pattern Analysis and Machine Intelligence*, 30(4):712–727, 2008.

[19] S. Boughorbel, J.P. Tarel, and F. Fleurel. Non-mercer kernels for svm object recognition. *British Machine Vision Conference*, pages 137–146, 2004.

[20] G.J. Burghouts and J.M. Geusebroek. Performance evaluation of local color invariants. *Computer Vision and Image Understanding*, 113(1):48–62, 2009.

[21] P. Carbonetto, N. de Freitas, and K. Barnard. A statistical model for general contextual object recognition. *Proc. European Conference on Computer Vision*, pages 350–36, 2004.

[22] G. Carneiro, A.B. Chan, P.J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE Trasactions on Pattern Analysis and Machine Intelligence*, 29(3):394–410, 2007.

[23] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[24] S.F. Chang, D. Ellis, W. Jiang, K. Lee, A. Yanagawa, A.C. Loui, and J. Luo. Large-scale multimodal semantic concept detection for consumer video. *ACM Internaltional Workshop on MIR*, pages 255–264, 2007.

[25] S.F. Chang, J.F. He, Y.G. Jiang, A. Yanagawa, E. Zavesky, E. Khoury, and C.W. Ngo. Columbia university/vireo-cityu/irit trecvid2008 high-level feature extraction and interactive video search. *NIST TRECVID workshop*, 2008. Gaithersburg, MD.

[26] S.F. Chang, W. Hsu, W. Jiang, L. Kennedy, D. Xu, A. Yanagawa, and E. Zavesky. Columbia university trecvid-2006 video search and high-level feature extraction. *NIST TRECVID workshop*, 2006. Gaithersburg, MD.

[27] S.F. Chang, W. Hsu, L. Kennedy, L. Xie, A. Yanagawa, E. Zavesky, and D. Zhang. Columbia university trecvid-2005 video search and high-level feature extraction. *NIST TRECVID workshop*, 2005. Gaithersburg, MD.

[28] N. Checka and K. Wilson. Person tracking using audio-video sensor fusion. *Technical Report*, 2002. MIT Artificial Intelligence Laboratory, Cambridge, MA.

[29] Y.X. Chen and J.Z. Wang. Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research*, 5:913–939, 2004.

[30] S. Chu and S. Narayanan. Environmental sound recognition using mp-based features. *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1–4, 2008.

[31] O. Chum and A. Zisserman. An exemplar model for learning object classes. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. Minneapolis, MN.

[32] D. Crandall and D. Huttenlocher. Composite models of objects and scenes for category recognition. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. Minneapolis, MN.

[33] M. Cristani, B. Manuele, and M. Vittorio. Audio-visual event recognition in surveillance video sequences. *IEEE Transactions on Multimedia*, 9(2):257–267, 2007.

[34] N. Cristianini, J. Taylor, A. Elisseeff, and J. Kandola. On kernel-target alignment. *Proc. Neural Information Processing Systems*, 2002.

[35] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 886–893, 2005.

[36] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. *European Conference on Computer Vision*, pages 7–13, 2006.

[37] J.W. Davis and A.F. Bobick. The representation and recognition of action using temporal templates. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 928–934, 1997.

[38] F. DelaTorre and O. Vinyals. Learning kernel expansions for image classification. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. Minneapolis, MN.

[39] D. Dementhon and D. Doermann. Video retrieval using spatial-temporal descriptors. *ACM International Conference on Multimedia*, 2003.

[40] Y. Deng and B.S. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):800–810, 2001.

[41] T.G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, randomization. *Machine Learning*, 40(2):139–157, 1998.

[42] S.K. Divvala, D. Hoiem, J.H. Hays, A.A. Efros, and M. Hebert. An empirical study of context in object detection. *Proc. IEEE Internationa Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[43] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 65–72, 2005.

[44] X. Dong and S.F. Chang. Visual event recognition in news video using kernel methods with multi-level temporal alignment. *Proc. IEEE Internationa Conference on Computer Vision and Pattern Recognition*, 2007. Minneapolis, MN.

[45] J. Driver. Enhancement of selective listening by illusory mislocation of speech sounds due to lip-reading. *Nature*, 381:66–68, 1996.

[46] S. Dupont and J. Luettin. Audio-visual speech modelling for continuous speech recognition. *IEEE Transactions on Multimedia*, 2:141–151, 2000.

[47] P. Duygulu, K. Barnard, and D.F.N. Freitas. Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. *Proc. European Conference on Computer Vision*, pages 97–112, 2002.

[48] S. Ebadollahi, L. Xie, S.F. Chang, and J. Smith. Visual event detection using multi-dimensional doncept dynamics. *Proc. IEEE International Conference on Multimedia and Expo*, pages 881–884, 2006.

[49] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing actions at a distance. *Proc. IEEE Internationa Conference on Computer Vision*, 2:726–733, 2003.

[50] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transaction on Image Processing*, 15(12):3736–3745, 2006.

[51] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge 2008 (voc2008) results. 2008. http://www.pascalnetwork.org/challenges/VOC/voc2008/workshop.

[52] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 60(2):1–36, 2009.

[53] P.F. Felzenszwalb and D.P. Huttenlocher. Pictorial structures for object recognition. *Internationa Journal on Computer Vision*, 61(1):55–79, 2005.

[54] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 264–271, 2003.

[55] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(1):36–51, 2008.

[56] V. Ferrari, F. Jurie, and C. Schmid. Accurate object detection with deformable shape models learnt from images. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. Minneapolis, MN.

[57] M. Fink and P. Perona. Mutual boosting for contextual inference. *Advances in Neural Information Processing Systems*, 28(22):337–407, 2000.

[58] J.W. Fisher and T. Darrell. Speaker association with signal-level audio visual fusion. *IEEE Transactions on Multimedia*, 6(3):406–413, 2004.

[59] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28(22):337–407, 2000.

[60] T.Y. Fu, X.X. Liu, L.H. Liang, X.B. Pi, and A.V. Nefian. Audio-visual speaker identification using coupled hidden markov models. *Proc. IEEE International Conference on Image Processing*, 3:29–32, 2003.

[61] C. Galleguillos, A. Rabinovich, and S. Belongie. Object categorization using co-occurrence, location and appearance. *Proc. IEEE International Conference on Computer Vision*, 2008. Anchorage, Alaska.

[62] J.M. Geusebroek, R. van den Boomgaard, A.W.M. Smeulders, and H. Geerts. Color invariance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1338–1350, 2001.

[63] S. Gould, J. Rodgers, D. Cohen, G. Elidan, and D. Koller. Multi-class segmentation with relative location prior. *International Journal on Computer Vision*, 80(3):300–316, 2008.

[64] K. Grauman and T. Darrel. The pyramid match kernel: Discriminative classification with sets of image features. *Proc. IEEE International Conference on Computer Vision*, 2:1458–1465, 2005.

[65] K. Grauman and T. Darrel. Approximate correspondences in high dimensions. *NIPS*, 2006.

[66] R. Gribonval and S. Krstulovic. Mptk, the matching pursuit toolkit. 2006. http://mptk.irisa.fr/.

[67] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. *Technical report, California Institute of Technology*, 2007.

[68] C.H. Gu, J.J. Lim, P. Arbeláez, and J. Malik. Recognition using regions. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[69] Y. Gutfreund, W. Zheng, and E.I. Knudsen. Gated visual input to the central auditory system. *Science*, 297(5586), 2002. 1556-1559.

[70] E. Hadjidemetriou, M. Grossberg, and B. Nayar. Multiresolution histograms and their use for recognition. *IEEE Trasactions on Pattern Analysis and Machine Intelligence*, 26(7):831–847, 2004.

[71] B. Han, D. Comaniciu, Z. Ying, and L. Davis. Incremental density approximation and kernel-based bayesian filtering for object tracking. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 638–644, 2004.

[72] D. Han, L.F. Bo, and C. Sminchisescu. Selection and context for action recognition. *Proc. IEEE International Conference on Computer Visionn*, 2009. Kyoto, Japan.

[73] C. Harris and M. Stephens. A combined corner and edge detector. *Alvey Vision Conference*, pages 147–151, 1988.

[74] X. He, R. Zemel, and M. Carreira-Perpinan. Multiscale conditional random fields for image labeling. *Proc. International Conference on Computer Vision and Pattern Recognition*, 2:695–702, 2004.

[75] X. He, R.S. Zemel, and D. Ray. Learning and incorporating top-down cues in image segmentation. *Proc. Europearn Conference on Computer Vision*, 1:338–351, 2006.

[76] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. *Proc. European Conference on Computer Vision*, pages 30–43, 2008.

[77] M.E. Hellman and J.Raviv. Probability of error, equivocation, and the chernoff bound. *IEEE Transactions on Information Theory*, 16(4):368–372, 1970.

[78] M.E. Hennecke, D.G. Stork, and K.V. Prasad. Visionary speech: Looking ahead to practical speechreading systems. *Speechreading by Humans and Machines. D.G. Stork and M.E. Hennecke, Eds. Berlin, Germany: Springer-Verlag*, pages 331–349, 1996.

[79] J. Hershey and J. Movellan. Audio-vision: Using audio-visual synchrony to locate sounds. *Advances in Neural Information Processing Systems*, 12:813–819, 1999.

[80] T. Hoffmann. Probabilistic latent semantic indexing. *Proc. International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57, 1999.

[81] S. Hoi, M.R. Lyu, and E.Y. Chang. Learning the unified kernel machines for classification. *Proc. ACM International Conference on Knowledge Discovery and Data Mining*, pages 187–196, 2006.

[82] D. Hoiem, A.A. Efros, and M. Hebert. Geometric context from a single image. *IEEE Proc. International Conference on Computer Vision*, pages 654–661, 2005.

[83] D. Hoiem, A.A. Efros, and M. Hebert. Putting objects in perspective. *IEEE Proc. International Conference on Computer Vision and Pattern Recognition*, 2:2137–2144, 2006.

[84] D. Hoiem, A.A. Efros, and M. Hebert. Recovering surface layout from an image. *International Journal on Computer Vision*, 75(1):151–172, 2007.

[85] A. Holub and P. Perona. A discriminative framework for modeling object class. *IEEE Proc. International Conference on Computer Vision and Pattern Recognition*, 1:664–671, 2005.

[86] P. Hough. Method and means for recognizing complex patterns. *U.S. Patent 3069654*, 1962.

[87] Y.X. Hu, L.L. Cao, F.J. Lv, S.C. Yan, Y.H. Gong, , and T. Huang. Action detection in complex scenes with spatial and temporal ambiguities. *Proc. IEEE International Conference on Computer Vision*, 2009. Kyoto, Japan.

[88] J. Huang. Color-spatial image indexing and applications. *PhD thesis, Cornell University*, 1998.

[89] J. Huang, Z. Liu, and Y. Wang. Joint scene classification and segmentation based on hidden markov model. *IEEE Transactions on Multimedia*, 7(3):538–550, 2005.

[90] J. Huang, Z. Liu, Y. Wang, Y. Chen, and E.K. Wong. Integration of multimodal features for video scene classification based on hmm. *Proc. IEEE Workshop on Multimedia Signal Processing*, pages 53–58, 1999.

[91] ImageCLEF. The clef cross language image retrieval track (imageclef). 2003 to 2009. http://ir.shef.ac.uk/imageclef/.

[92] K. Iwano, T. Yoshinaga, S. Tamura, and S. Furui. Audio-visual speech recognition using lip information extracted from side-face images. *EURASIP Journal on Audio, Speech, and Music Processing*, 2007(1):4–4, 2007.

[93] M. Izumi and A. Kojiama. Generating natural language description of human behavior from video images. *Proc. IEEE International Conference on Pattern Recognition*, 4:728–731, 2000.

[94] A. Jepson, D.J. Fleet, and T.F. El-Maraghi. Robust online appearence models for visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1296–1311, 2003.

[95] H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. *Proc. IEEE International Conference on Computer Vision*, 2007. Janeiro, Brazil.

[96] W. Jiang, S.F. Chang, and A.C. Loui. Active context-based concept fusion with partial user labels. *Proc. IEEE International Conference on Image Processing*, pages 2917–2920, 2006.

[97] W. Jiang, S.F. Chang, and A.C. Loui. Context-based concept fusion with boosted conditional random fields. *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 1:949–952, 2007.

[98] W. Jiang, S.F. Chang, and A.C. Loui. Kernel sharing with joint boosting for multi-class concept detection. *Proc. IEEE International Workshop on Semantic Learning Applications in Multimedia, in conjuction with International Confence on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[99] W. Jiang, C. Cotton, S.F. Chang, D. Ellis, and A.C. Loui. Short-term audio-visual atoms for generic video concept classification. *Proc. ACM International Confence on Multimedia*, pages 5–14, 2009.

[100] W. Jiang, H. Er, Q. Dai, and J. Gu. Similarity-based online feature selection in content-based image retrieval. *IEEE Transactions on Image Processing*, 15(3):702–712, 2006.

[101] Y.G. Jiang, C.W. Ngo, and J. Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. *ACM International Conference on Image and Video Retrieval*, 2007. Amsterdam, The Netherlands.

[102] Y.G. Jiang, J. Wang, S.F. Chang, and C. Ngo. Domain adaptive semantic diffusion for large scale context-based video annotation. *Proc. IEEE International Conference on Computer Vision*, 2009. Kyoto, Japan.

[103] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. *Proc. IEEE International Conference on Computer Vision*, 1:604–610, 2005.

[104] T. Kadir and M. Brady. Saliency, scale and image description. *International Journal on Computer Vision*, 45(2):83–105, 2001.

[105] B. Kapralos, M.R.M. Jenkin, and E. Milios. Audiovisual localization of multiple speakers in a video teleconferencing settings. *International Journal of Imaging Systems and Technology*, 13:95–105, 2003.

[106] R. Kaucic, B. Dalton, and A. Blake. Real-time lip tracking for audio-visual speech recognition applications. *Proc. European Conference on Computer Vision*, 2:376–387, 1996.

[107] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. *Proc. IEEE International Conference on Computer Vision*, 2007. Janeiro, Brazil.

[108] J. Kittler, M. Hatef, R.P.W. Duin, and J. Mates. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[109] P. Kohli, L. Ladický, and P. Torr. Robust higher order potentials for enforcing label consistency. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[110] A. Kumar and C. Sminchisescu. Support kernel machines for object recognition. *Proc. International Conference on Computer Vision*, 2007. Janeiro, Brazil.

[111] S. Kumar and M. Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. *Proc. International Conference on Computer Vision*, pages 1150–1157, 2003.

[112] S. Kumar and M. Hebert. A hierarchical field framework for unified context-based classification. *Proc. International Conference on Computer Vision*, pages 1150–1157, 2005.

[113] M. Kyperountas, C. Kotropoulos, and I. Pitas. Enhanced eigen-audioframes for audiovisual scene change detection. *IEEE Transactions on Multimedia*, 9(4):785–797, 2007.

[114] J. Lafferty, A. M. Fernando, and C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proc. International Conference on Machine Learning*, pages 282–289, 2001.

[115] C.H. Lampert, M.B. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[116] G.R.G. Lanckriet, N. Cristianini, P. Bartlett, L.E. Ghaoui, and M.I. Jordan. Learning the kernel matrix with semidefinite programming. *The Journal of Machine Learning Research.*

[117] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[118] S. Lazebnik, C. Schmid, and J. Ponce. Sparse texture representation using affine-invariant neighborhood. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 319–324, 2003.

[119] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: spatial pyramid matching for recognizing natual scene categories. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2:2169–2178, 2006.

[120] A. Lehmann, B. Leibe, and L.J. Van gool. Prism: Principled implicit shape model. *British Machine Vision Conference*, 2009. London, UK.

[121] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. *ECCV Workshop on Statistical Learning in Computer Vision*, 2004.

[122] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal on Computer Vision*, 77(1-3):259–289, 2008.

[123] D. Li, N. Dimitrova, M. Li, and I.K. Sethi. Multimedia content processing through cross-modal association. *Proc. ACM International Conference on Multimedia*, pages 604–611, 2003.

[124] F. Li, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *IEEE CVPR Workshop on Generative-Model Based Vision*, 2004.

[125] F. Li and P. Perona. A bayesian hierarchical model for learning natural scene categories. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2:524–531, 2005.

[126] J. Li and J. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1075–1088, 2003.

[127] J. Li, W. Wu, T. Wang, and Y. Zhang. One step beyond histogram: Image representation using markov stationary features. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[128] L.J. Li and F.F. Li. What, where and who? classifying events by scene and object recognition. *Proc. IEEE International Conference on Computer Vision*, 2007. Janeiro, Brazil.

[129] L.J. Li, R. Socher, and F.F. Li. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[130] W.H. Lin and A. Hauptmann. News video classification using svm-based multimodal classifiers and combination strategies. *Proc. ACM International Conference on Multimedia*, pages 323–326, 2002.

[131] W.H. Lin, R. Jin, and A. Hauptmann. Meta-classification of multimedia classifiers. *International Workshop on Knowledge Discovery in Multimedia and Complex Data*, 2002. Taipei, Taiwan.

[132] T. Lindeberg. Scale-space. *Encyclopedia of Computer Science and Engineering (Benjamin Wah, ed), John Wiley and Sons*, 4:2495–2504, 2009.

[133] T. Lindeberg and J. Garding. Shape-adapted smoothing in estimation of 3-d shape cues from affine deformations of local 2-d brightness structure. *Image and Vision Computing*, 15(6):415–434, 1997.

[134] H.B. Ling and S. Soatto. Proximity distribution kernels for geometric context in category recognition. *Proc. IEEE International Conference on Computer Vision*, 2007. Janeiro, Brazil.

[135] J. Liu, S. Ali, and M. Shah. Recognizing human actions using multiple features. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[136] S. Liu, H. Yi, L.T. Chia, and D. Rajan. Adaptive hierarchical multi-class svm classifier for texture-based image classification. *Proc. IEEE International Conference on Multimedia and Expo*, page 4, 2005.

[137] A. Loui, J. Luo, S.F. Chang, D. Ellis, W. Jiang, L. Kennedy, K. Lee, and A. Yanagawa. Kodak's consumer video benchmark data set: concept definition and annotation. *ACM SIGMM International Workshop on MIR*, pages 245–254, 2007.

[138] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[139] LSCOM. Lscom lexicon definitions and annotations version 1.0. *DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia, Columbia University ADVENT Technical Report 217-2006-3*, March 2006.

[140] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proc. Imaging understanding workshop*, pages 121–130, 1981.

[141] S. Lucey, T. Chen, S. Sridharan, and V. Chandran. Integration strategies for audio-visual speech processing: Applied to text-dependent speaker recognition. *IEEE Transactions on Multimedia*, 7(3):495–506, 2005.

[142] J. Luo, B. Caputo, A. Zweig, J.H. Bach, and J. Anemüller. Object category detection using audio-visual cues. *Proc. International Conference on Computer Vision Systems*, 2008. Santorini, Greece.

[143] F. Mahmoudi, J. Shanbehzadeh, A.M. Eftekhari-Moghadam, and H. Soltanian-Zadeh. Image retrieval based on shape similarity by edge orientation autocorrelogram. *Pattern Recognition*, (2):1725–1736, 2003.

[144] S. Maji, A. Berg, and J. Malik. Classification using intersection kernel support vector machine is efficient. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[145] S. Maji and J. Malik. Object detection using a max-margin hough transform. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[146] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

[147] M.I. Mandel, G.E. Poliner, and D. Ellis. Support vector machine active learning for music retrieval. *Multimedia systems*, 12(1):3–13, 2006.

[148] B.S. Manjunath and W.Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.

[149] B.S. Manjunath, J.R. Ohm, V.V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):703–715, 2001.

[150] O. Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. *NIPS*, pages 570–576, 1998.

[151] M. Marszałek, I. Laptev, and C. Schmid. Actions in context. *Proc. International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[152] J. Matas, O. Chum, M. Urba, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. *Proc. British Machine Vision Conference*, pages 384–396, 2002.

[153] K. Mikolajcyk and C. Schmid. An affine invariant interest point detector. *Proc. International Conference on Computer Vision*, 2002. Vancouver, Canada.

[154] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 1(60):63–86, 2004.

[155] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(27):1615–1630, 2005.

[156] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005.

[157] P. Montesinos, V. Gouet, and R. Deriche. Differential invariants for color images. *Proc. International Conference on Pattern Recognition*, pages 838–840, 1998.

[158] F. Moosmann, B. Triggs, and F. Jurie. Randomized clustering forests for building fast and discriminative visual vocabularies. *Advances in Neural Information Processing Systems*, 2007.

[159] S. Munder and D. M. Gavrila. An experimental study on pedestrian classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1863–1868, 2006.

[160] D. Munoz, J.A. Bagnell, N. Vandapel, and M. Hebert. Contextual classification with functional max-margin markov networks. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[161] K. Murphy, A. Torralba, and W. T. Freeman. Using the forest to see the trees: a graphical model relating features, objects, and scenes. *NIPS*, 2003.

[162] M. Naphade, I.V. Kozintsev, and T.S. Huang. A factor graph framework for semantic video indexing. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(1):40–52, 2002.

[163] M. Naphade, J. R. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-scale concept ontology for multimedia. *IEEE MultiMedia*, 13:86–91, 2006.

[164] M. R. Naphade, L. Kennedy, J. R. Kender, S.-F. Chang, J. R. Smith, P. Over, and A. Hauptmann. A light scale concept ontology for multimedia understanding for trecvid 2005. *IBM Research Technical Report*, 2005.

[165] R. Polanaand R. Nelson. Detecting activities. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 2–7, 1993.

[166] B. Ni, S. Yan, and A. Kassim. Contextualizing histogram. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[167] W. Niblack, R. Barber, W. Equitz, M. Fickner, E. Glasman, D. Petkovic, and P. Yanker. The qbic project: Querying images by content using color, texture and shape. *SPIE Conference on Geometric Methods in Computer Vision*, 1908:173–187, 1993.

[168] J.C. Niebles, B. Han, A. Ferencz, and F.F. Li. Extracting moving people from internet videos. *European Conference on Computer Vision*, pages 527–540, 2008.

[169] J.C. Niebles, H. Wang, and F.F. Li. Unsupervised learning of human action categories using spatial-temporal words. *International Journal on Computer Vision*, 79(3):299–318, 2008.

[170] NIST. Trec video retrieval evaluation (trecvid). 2001 to 2008. http://www-nlpir.nist.gov/projects/trecvid/.

[171] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. *European Conference on Computer Vision*, pages 490–503, 2006.

[172] J. Ogle and D. Ellis. Fingerprinting to identify repeated sound events in long-duration personal audio recordings. *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pages I–233–236, 2007.

[173] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 24(7):971–987, 2002.

[174] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *Internal Journal of Computer Vision*, 42(3):145–175, 2001.

[175] Björn Ommer and J. Malik. Multi-scale object detection by clustering lines. *Proc. IEEE International Conference on Computer Vision*, 2009. Kyoto, Japan.

[176] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer. Generic object recognition with boosting. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 28(3):416–431, 2006.

[177] A. Opelt, A. Pinz, and A. Zisserman. Incremental learning of object detectors using a visual shape alphabet. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 1:3–10, 2006.

[178] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 130–136, 1997.

[179] S. Paek and S.F. Chang. Experiments in constructing belief networks for image classification systems. *Proc. IEEE International Conference on Image Processing*, 3:46–49, 2000.

[180] C. Papageorgiou and T. Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1), 2000.

[181] N. Paragios and V. Ramesh. A mrf-based real-time approach for subway monitoring. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 1034–1040, 2001.

[182] PASCAL. The pascal visual object classes challenge. 2005 to 2009. http://pascallin.ecs.soton.ac.uk/challenges/VOC/.

[183] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. *Proc. ACM International Conference on Multimedia*, pages 65–73, 1996.

[184] H. Permuter, J. Francos, and I. Jermyn. A study of gaussian mixture models of color and texture features for image classification and segmentation. *Pattern Recognition*, 39(4):695–706, 2006.

[185] N. Peterfreund. Robust tracking of position and velocity with kalman snakes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(6):564–569, 1999.

[186] F. Petitcolas. Mpeg for matlab. 2003. http://www.petitcolas.net/fabien/software/mpeg.

[187] N. Pinto, D.D. Cox, and J.J. DiCarlo. Why is real-world visual object recognition hard. *PLoS Computational Biology*, 4(1):151–156, 2008.

[188] R. Pless. Using many cameras as one. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 587–593, 2003.

[189] R. Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, 2006.

[190] M. Pontil and A. Verri. Support vector machines for 3d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(6):637–646, 1998.

[191] G. Potamianos, C. Neti, G. Gravier, A.Garg, and A.W. Senior. Recent advances in the automatic recognition of audio visual speech. *Proceedings of the IEEE*, 91(9):1306–1326, 2003.

[192] G.J. Qi, X.S. Hua, Y. Rui, J. Tang, T. Mei, and H.J. Zhang. Correlative multi-label video annotation. *Proc. ACM International Conference on Multimedia*, pages 883–890, 2007.

[193] P. Quelhas, F. Monay, J. Odobez, D.G.P.T. Tuytelaars, and L.V. Gool. Modeling scenes with local descriptors and latent aspects. *Proc. IEEE International Conference on Computer Vision*, pages 883–890, 2005.

[194] A. Rabinovich, S. Belongie, T. Lange, and J.M. Buhmann. Model order selection and cue combination for image segmentation. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 1:1130–1137, 2006.

[195] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. *Proc. IEEE International Conference on Computer Vision*, 2007. Janeiro, Brazil.

[196] D. Ramanan, D.A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1), 2007.

[197] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *International Journal on Computer Vision*, 50(2), 2002.

[198] N. Rasiwasia and N. Vasconcelos. Holistic context modeling using semantic co-occurrences. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[199] 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. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2:1605–1614, 2006.

[200] P.A. Sandon. Simulating visual attention. *Journal of Cognitive Neuroscience*, 2(3):213–231, 1990.

[201] S. Savarese, A. DelPozo, J.C. Niebles, and F.F. Li. Spatial-temporal correlations for unsupervised action classification. *IEEE Workshop on Motion and Video Computing*, 2008. Copper Mountain, Colorado.

[202] R. Schapire and Y. Singer. Improved boosting algorithm using confidence-rated predictions. *Proc. Annual Conference on Computational Learning Theory*, pages 80–91, 1998.

[203] C. Schauer and H.M. Gross. A computational model of early auditory-visual integration. *Pattern Recognition*, 2781:362–369, 2003.

[204] B. Schiele and J. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1), 2000.

[205] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. *Proc. IEEE International Conference on Pattern Recognition*, 3:32–36, 2004.

[206] Y. Sheikh, M. Sheikh, and M. Shah. Exploring the space of a human action. *Proc. IEEE International Conference on Computer Vision*, 1:144– 149, 2005.

[207] J. Shi and C. Tomasi. Good features to track. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[208] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. *Proc. IEEE International Conference on Computer Vision*, 1:503–510, 2005.

[209] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[210] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Texton-boost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. *European Conference on Computer Vision*, pages 1–15, 2006.

[211] A. Singhal, J. Luo, and W. Zhu. Probabilistic spatial context models for scene content understanding. *Proc. IEEE International Conference on Computer Vision*, 1:235–241, 2003.

[212] M. Slaney and M. Covell. Facesync: A linear operator for measuring synchronization of video facial images and audio tracks. *Advances in Neural Information Processing Systems*, 13:814–820, 2000.

[213] P. Smaragdis and M. Casey. Audio/visual independent components. *International Symposium on Independent Component Analysis and Blind Source Separation*, pages 709–714, 2003.

[214] J. Smith, C. Lin, M. Naphade, A. Natsev, and B. Tseng. Multimedia semantic indexing using model vectors. *Proc. IEEE International Conference on Multimedia & Expo*, 2:445–448, 2003.

[215] C.G.M. Snoek, J.C. van Gemert, J.M. Geusebroek, B. Huurnink, D.C. Koelma, G.P. Nguyen, O. de Rooij, F.J. Seinstra, A.W.M. Smeulders, C.J. Veenman, and M. Worring. The mediamill trecvid 2005 semantic video search engine. *Proc. NIST TRECVID Workshop*, Gaithersburg, USA 2005.

[216] C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757, 2002.

[217] M.A. Stricker and M. Orengo. Similarity of color images. *Proc. SPIE Storage and Retrieval for Image and Video Databases*, 2420:381–392, 1995.

[218] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky. Learning hierarchical models of scenes, objects, and parts. *Proc. IEEE International Conference on Computer Vision*, pages 1331–1338, 2005.

[219] K. Tieu and P. Viola. Boosting image retrieval. *International Journal of Computer Vision*, 56:228–235, 2000.

[220] A. Torralba. Contextual priming for object detection. *International Journal on Computer Vision*, 53(2):169–191, 2003.

[221] A. Torralba, K. Murphy, W. Freeman, and M. Rubin. Context-based vision system for place and object recognition. *Proc. IEEE International Conference on Computer Vision*, 1:273–280, 2003.

[222] A. Torralba, K. Murphy, and W.T. Freeman. Contextual models for object detection using boosted random fields. *NIPS*, 2004.

[223] A. Torralba, K. Murphy, and W.T. Freeman. Sharing features: effective boosting procedure for multiclass object detecion. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2:762–769, 2004.

[224] Z. Tu. Auto-context and its application to high-level vision tasks. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[225] Z. Tu, X. Chen, A. Yuille, and S. Zhu. Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision*, 63(2):113–140, 2005.

[226] T. Tuytelaars and K. Mikolajczyk. *Local invariant feature detectors: A survey.* 2008. Now Publishers Inc.

[227] A. Vailaya, A. Jain, and H.J. Zhang. On image classification: City vs. landscapes. *Pattern Recognition*, 31(12):1921–1935, 1998.

[228] K.E.A. van de Sande, T. Gevers, and C.G.M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010. (in press).

[229] J. van de Weijer, T. Gevers, and A.D. Bagdanov. Boosting color saliency in image feature detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):150–156, 2006.

[230] V. Vapnik. Statistical learning theory. *Wiley-Interscience, New York*, 1998.

[231] N. Vasconcelos. Feature selection by maximum marginal diversity: optimality and implications for visual recognition. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 1:762–769, 2003.

[232] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. *IEEE International Conference on Computer Vision*, 2009. Kyoto, Japan.

[233] J. Vermaak, M. Gangnet, A. Blake, and P. Perez. Sequential monte carlo fusion of sound and vision for speaker tracking. *Proc. IEEE International Conference on Computer Vision*, 1:741–746, 2001.

[234] P.A. Viola and M.J. Jones. Robust real-time face detection. *International Journal on Computer Vision*, 57(2):137–154, 2004.

[235] C. Wallraven. Recognition with local features: the kernel recipe. *Proc. IEEE International Conference on Computer Vision*, 1:257–264, 2003.

[236] X.G. Wang, K. Tieu, and E. Grimson. Learning semantic scene models by trajectory analysis. *European Conference on Computer Vision*, pages 110–123, 2006.

[237] T. Wark, S. Sridharan, and V. Chandran. The use of temporal speech and lip information for multi-modal speaker identification via multi-stream hmms. *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2389–2392, 2000.

[238] M.F. Weng and Y.Y. Chuang. Multi-cue fusion for semantic video indexing. *Proc. ACM International Conference on Multimedia*, pages 71–80, 2008.

[239] J. Winn and N. Jojic. Locus: Learning object classes with unsupervised segmentation. *Proc. IEEE International Conference on Computer Vision*, 1:756–763, 2005.

[240] J. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 1:37–44, 2006.

[241] L. Wolf and S. Bileschi. A critical view of context. *International Journal on Computer Vision*, 69(2):251–261, 2006.

[242] S. Wong, T. Kim, and R. Cipolla. Learning motion categories using both semantic and structural information. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. Minneapolis, MN.

[243] M. Worring, C. Snoek, O. de Rooij, G.P. Nguyen, and A. Smeulders. The mediamill semantic video search engine. *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 4:1213–1216, 2007.

[244] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. Rehg. A scalable approach to activity recognition based on object use. *Proc. IEEE International Conference on Computer Vision*, 2007. Janeiro, Brazil.

[245] Y. Wu, C.Y. Lin, E.Y. Chang, and J.R. Smith. Multimodal information fusion for video concept detection. *Proc. IEEE International Conference on Image Processing*, pages 2391–2394, 2004.

[246] Y. Wu, B.L. Tseng, and J.R. Smith. Ontology-based multi-classification learning for video concept detection. *Proc. IEEE International Conference on Multimedia and Expo*, 2:1003–1006, 2004.

[247] R. Yan, M. Chen, and A. Hauptmann. Mining relationship between video concepts using probabilistic graphical models. *Proc. IEEE International Conference on Multimedia and Expo*, pages 301–304, 2006.

[248] A. Yanagawa, S.F. Chang, L. Kennedy, and W. Hsu. Columbia university's baseline detectors for 374 lscom semantic visual concepts. *Columbia University ADVENT Tech. Report 222-2006-8*, March 2007.

[249] A. Yanagawa, W. Hsu, and S.-F. Chang. Brief descriptions of visual features for baseline trecvid concept detectors. *Columbia University ADVENT Tech. Report 219-2006-5*, July 2006.

[250] A. Yanagawa, A. Loui, J. Luo, S.F. Chang, D. Ellis, W. Jiang, L. Kennedy, and K. Lee. Kodak consumer video benckmark data set: concept definition and annotation. *Columbia University ADVENT Technical Report 222-2008-8*, September 2008.

[251] C. Yang, M. Dong, and J. Hua. Region-based image annotation using asymmetrical support vector machine-based multiple-instance learning. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, pages 2057–2063, 2006.

[252] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching. *Proc. IEEE International Conference on Computer Vision*, 2009. Kyoto, Japan.

[253] L. Yang, R. Jin, R. Sukthankar, and F. Jurie. Unifying discriminative visual codebook generation with classifier training for object category recognition. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. Anchorage, Alaska.

[254] Y. Yemez, A. Kanak, E. Erzin, and A.M. Tekalp. Multimodal speaker identification with audio-video processing. *Proc. IEEE International Conference on Image Processing*, 3:5–8, 2003.

[255] J. Yuan, Z. Liu, and Y. Wu. Discriminative subvolume search for efficient action detection. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. Miami, Florida.

[256] L. Zelnik-Manor and M. Irani. Statistical analysis of dynamic actions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1530–1535, 2006.

[257] D. Zhang and S.F. Chang. A generative-discriminative hybrid method for multi-view object detection. *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2:2017–2024, 2006.

[258] H. Zhang, A. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. *IEEE International Conference on Computer Vision and Pattern Recognition*, 2:2126–2136, 2006.

[259] 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 on Computer Vision*, 73(2):213–238, 2007.

[260] G.Q. Zhao, L. Chen, J. Song, and G. Chen. Large head movement tracking using sift-based registration. *ACM International Conference on Multimedia*, 2007.

[261] W.S. Zheng, S.G. Gong, and T. Xiang. Quantifying contextual information for object detection. *Proc. IEEE International Conference on Computer Vision*, Kyoto, Japan 2009.

[262] H. Zhou, Y. Yuan, and C. Shi. Object tracking using sift features and mean shift. *Computer Visision and Image Understanding*, 113(3):345–352, 2009.

[263] X. Zhou, X. Zhuang, S. Yan, S.F. Chang, M. Hasegawa-Johnson, and T.S. Huang. Sift-bag kernel for video event analysis. *Proc. ACM international conference on Multimedia*, pages 229–238, 2008.

[264] X. Zhu, J. Kandola, Z. Ghahramani, and J. Lafferty. Nonparametric transforms of graph kernels for semi-supervised learning. *Advances in Neural Information Processing Systems*, 2005.

[265] D. Zotkin, R. Duraiswami, and L. Davis. Joint audio-visual tracking using particle filters. *EURASIP Journal on Applied Signal Processing*, 11:1154–1164, 2002.

[266] X. Zou and B. Bhanu. Tracking humans using multi-modal fusion. *Proc. IEEE International Conference on Computer Vision and Pattern Recognitiong*, 3:4, 2005.