Towards Autonomous Situation Awareness

by

Nikhil Naikal

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Electrical Engineering and Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Shankar S. Sastry, Chair
Professor Ruzena Bajcsy
Professor Bruno Olshausen

Spring 2014

Towards Autonomous Situation Awareness

Abstract

Towards Autonomous Situation Awareness

by

Nikhil Naikal

Doctor of Philosophy in Electrical Engineering and Computer Science

University of California, Berkeley

Professor Shankar S. Sastry, Chair

Technological advances in communication and computing coupled with low costs of sensors have enabled large-scale deployment of video camera networks in our environment nowadays. They are increasingly being used by decision-makers for perceiving elements of an environment in real-time, comprehending their meaning and predicting their status in the near future. Such situational awareness is crucial in dynamic scenarios that arise in military command and control, facility security and emergency services such as policing and fire-fighting. While human perception driven situation awareness has worked well in constrained settings, it is unfortunately not scalable. Furthermore, scenarios such as security and surveillance commonly involve highly complex cognitive tasks that can quickly become monotonous and mentally taxing for human operators. In this thesis, we present new frameworks for automating the perception stage of situation awareness.

We begin this thesis with the development of a system that is capable of categorizing objects and landmarks in an efficient and distributed manner. Our system is designed to operate with networks of wireless smart cameras for local perception, and a central station for global inference. We demonstrate that this decoupling of the algorithm pipeline can drastically minimize the power and bandwidth consumed by the wireless cameras. Further, we experimentally validate that our multiple-view inference framework can significantly improve the performance of object and landmark categorization over traditional single-view settings.

In the second part of the thesis we extend our distributed object categorization framework to address the problem of automatic human activity detection and categorization. We are particularly interested in the development of rich representations for human motion that are invariant to perspective, scale and the speed at which actions are performed. We propose a generalized framework to perform spatiotemporal fusion of dynamic imagery from multiple wireless smart cameras, and validate the efficacy of our fusion framework on both publicly available, and novel datasets.

In most realistic scenarios that require situation awareness, objects and people occur in cluttered scenes and exhibit immense variability in their appearance, shape and pose. In the final part of this thesis, we analyze the interplay between computer vision tasks such as segmentation and categorization and present joint frameworks that significantly improve the performance of each task.

Our experimental analysis demonstrates that detection and categorization hypotheses help provide good segmentation results, and that segmentation can be used to prune errors in the hypothesis.

To my loving Parents for their constant support and encouragement.

Contents

List of Figures

List of Tables

Acknowledgments

I would like to begin by thanking my advisor, Prof. Shankar Sastry, for taking me on as a graduate student five years ago and providing me with the opportunity of undertaking this academic journey under his excellent guidance and support. Prof. Sastry's advice and feedback have been instrumental in improving the quality of my work as well as providing inspiration for new research directions. I am indebted to him for his keen eye for technical detail and his emphasis on mathematical rigor in our meetings. Prof. Sastry encouraged thoughtful discussions amongst all his students, which prepared us to exchange and analyze ideas in both academic and non-academic settings. I would like to thank him for allowing me to collaborate with various researchers, both inside and outside academia, for providing a comfortable learning environment, and for his counsel on academic and non-academic matters.

I would like to thank Prof. Ruzena Bajcsy for all her help and advice over the years. I had the pleasure of collaborating with Prof. Bajcsy for a large part of my graduate studies and her suggestions have always brought a fresh perspective to my research. I am immensely inspired by her passion and dedication to science. Having her on my thesis committee was extremely valuable to me, as she constantly engaged me in interesting discussions and encouraged me to explore new research directions. I would like to thank Prof. Bruno Olshausen for being a part of my thesis committee. His comments and suggestions from a neuroscience perspective helped me step back and reconsider the big picture.

Over the years, I have had the pleasure of working with some exceptional researchers. I would like to thank Dr. Allen Yang and Dr. Dheeraj Singaraju. It was a rare opportunity to collaborate with them and is definitely something I will miss. Our fruitful discussions not only led to the existence of this thesis but also deepened my understanding of the fundamentals of signal processing and computer vision. I am grateful to them for providing me with such excellent learning opportunities and for their patience in the process.

My groupmates provided invaluable help and support throughout graduate school. I would like to thank Henrik Ohlsson, Sourabh Amin, Humberto Gonzalez, Daniel Calderone, Ehsan Elhamifar, Lillian Ratliff, Dorsa Sadigh and Chi-Pang Lam. I am especially indebted to Sam Burden for encouraging me to work with Prof. Sastry and for helping me with my transition into his research group. I would also like to thank Jessica Gamble for all the help and support that she has offered during the past few years.

I'd like to acknowledge friends outside of Prof. Sastry's research group, namely - Ricardo Garcia, Meena Natarajan, Matthew Spencer, John Kua and Anil Aswani. Their support and encouragement made even the slow times of research enjoyable. I'd especially like to thank Brandy Lipton for being a constant pillar of support, and encouraging me to complete my thesis in a timely manner.

Finally, I'd like to thank my mom, Dr. Vani Santosh, for her patience and care and my dad, Dr. Santosh Kumar Naikal, for his humor and wisdom. I thank them for their unconditional love and encouragement, especially throughout the graduate school years.

Chapter 1

Introduction

1.1 Motivation

Situation awareness is the process of perceiving elements of an environment, comprehending their meaning, and predicting their status in the near future. It has primarily been studied in military theory where decision makers must deal with human performance in tasks that are physical or perceptual, as well as consider human behavior involving highly complex cognitive tasks with increasing frequency. Such tasks commonly arise in air traffic control, ship navigation, manufacturing systems operation, power plant operation, military command and control and emergency services such as fire fighting and policing. Many other everyday activities such as driving in heavy traffic, operating heavy machinery and medical procedures also call for a dynamic update of the situation for effective decision making. Thus, situation awareness serves as a means to close the loop between new information and existing knowledge in order to maintain a composite picture of any situation.

Situation awareness is typically broken down into three stages - perception, comprehension and projection [26]. The primary stage of perception involves the processes of monitoring, cue detection and simple recognition, which lead to an awareness of multiple situational elements (i.e. objects, events, people) and their current states (i.e. locations, conditions, actions). An air force tactical commander, for instance, might need accurate data on the location, type, number, capabilities and dynamics of all enemy and friendly forces in a given area and their relationship to other points of reference. The secondary comprehension stage involves a synthesis and integration of disjointed pieces of perceived information through the processes of pattern recognition, interpretation and evaluation in order to understand how it impacts goals and objectives. For example, the tactical commander must comprehend that the appearance of three enemy aircrafts within a certain proximity of one another indicates certain things about their objectives. Finally, the ability to project the future actions of the elements in the environment - at least in the vey near term - forms the third projection stage of situation awareness. This is achieved through knowledge of the status and dynamics of the elements and comprehension of the situation. For example, knowing that a threat aircraft is currently offensive and is in a certain location allows a tactical commander

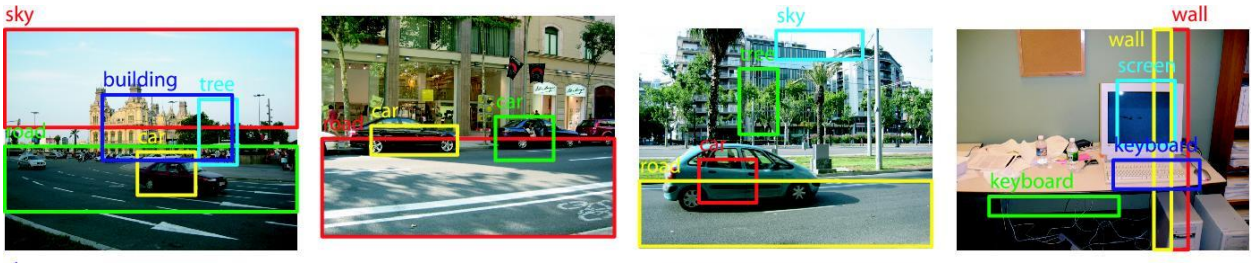


Figure 1.1: Detecting and categorizing objects in the scene

Figure 1.2: Two pairs of images with the same person in each pair exhibiting pose variability in the image pair.

to project that the aircraft is likely to attack in a given manner.

While human perception driven situational awareness has worked well in constrained settings, it is unfortunately not scalable. In today's world, we are faced with a seemingly ever-increasing human population who have a strong desire for instant information, automation and a quickly diminishing tolerance of failure. This has led to a growing demand for autonomous and reliable situational awareness systems in both public and private places composed of networks of smart sensors and actuators for perception, and centralized algorithms with human decision makers in-the-loop for comprehension and projection.

Video cameras are arguably the most useful sensors for perception. They are cheap to manufacture and robustly sense both small and large environments under varying lighting conditions. The low cost of video cameras coupled with the decreasing cost of network communication and computing have made automated perception a core research topic in the disciplines of computer vision and signal processing. In the context of computer vision, perception is studied as the following interesting and challenging sub-topics:

1. **Object detection and categorization**: This is the task of finding objects in an image or video sequence and identifying their semantic categories, e.g., cars, bikes, trees, etc. as shown in figure 1.1. Detecting and categorizing objects is a challenging problem because of their appearance variability due to differences in view-point, size, scale, translation and rotation. Objects could also be partially occluded from the field of view of the camera.

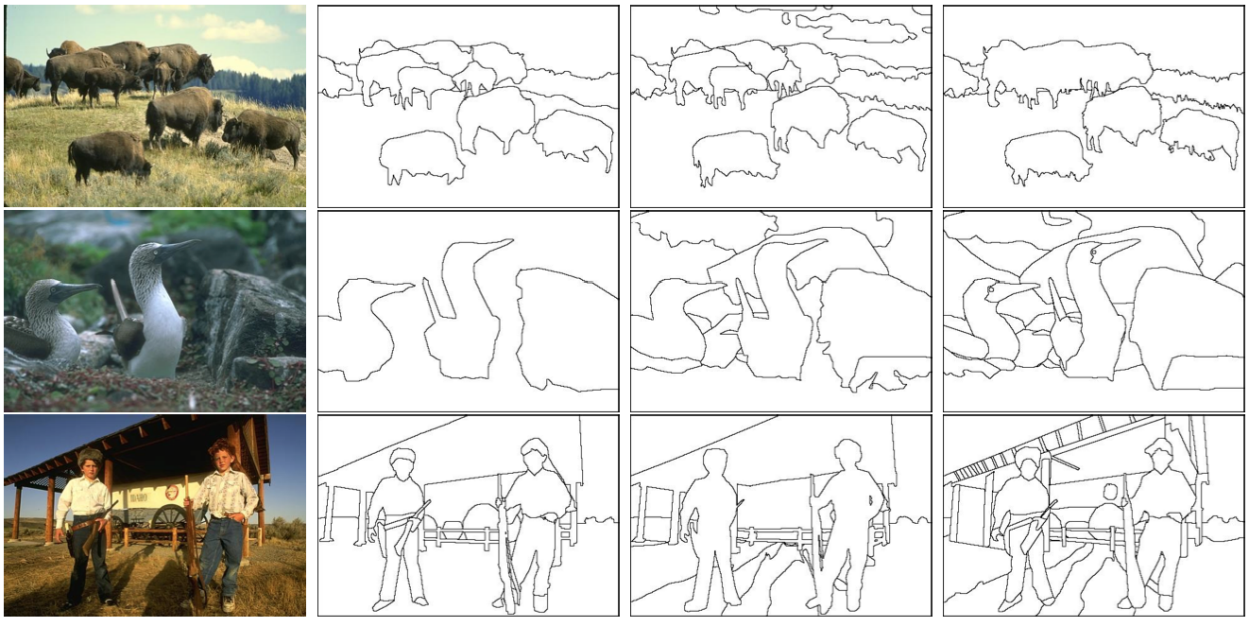


Figure 1.3: First column shows the query images to be segmented and the next three columns show example segmentations given by three different humans. Notice that for each image, the segmentations given by different people exhibit several differences.

2. **Human detection**: Human detection is an essential and significant task in any scenario that requires reliable localization of a person in camera footage. It is typically studied under two areas of research, namely - face detection and holistic person detection. Face detection is also studied along with face recognition in certain scenarios. Human detection is arguably a harder problem when compared to static object detection because a single person can himself exhibit a wide set of pose configurations as shown in figure 1.2.

3. **Segmentation and tracking**: Segmentation is the process of grouping together pixels in an image corresponding to a particular semantic category. Segmentation is generally considered as an ill-posed problem as even humans demonstrate a large variability in defining true region boundaries in images. This can be observed in figure 1.3 where different human annotators provide slightly varying segmentations for the same image. A closely related problem to segmentation is tracking. The objective of tracking is to predict the location of the pixels corresponding to a particular segmentation, in successive frames of a video sequence. Tracking can be very challenging in certain scenarios where the objects being tracked can get occluded from a camera's field-of-view.

4. **Gesture and activity recognition**: Gesture and activity recognition is the process of analyzing a human's body language. Unlike object and human detection, most gesture and activity recognition algorithms rely on the temporal information in video streams in order to disambiguate different human gestures. This can be a challenging problem because of the subtle

variations in the the rate and kinematics of different people performing the same actions.

Robust and scalable solutions to these sub-topics can enable autonomous situational awareness in a wide variety of settings. For instance, networks of wireless smart cameras can be a valuable tool for *security and surveillance* applications. They can be used to automatically detect what people are doing in public spaces, track and recognize objects that have been removed from or introduced into the scene, and alert authorities of perpetrators of crime while requiring fewer personnel to monitor raw camera imagery. Annotating raw camera footage with actions of people can also aid in content based retrieval of security camera footage. In the *search and rescue* context, wireless networks of cameras mounted on the search crew, and on unmanned aerial vehicles can assist in perceiving the environment and locating missing individuals. *Modern hospitals and elderly care centers* could utilize smart cameras to monitor the health and activities of patients and alert the nursing staff when a patient needs assistance. They can also be used to track the movement of health care professionals and patients within a hospital in order to map and mitigate spread of diseases.

In *modern built environments* such as smart offices and smart homes, gesture based control of appliances can increase convenience for inhabitants, and improve their productivity. Categorizing the objects in these environments can also help in easily tracking down items such as fire extinguishers and defibrillators during emergency situations. In urban settings, detecting and tracking pedestrians and bicyclists within an *intelligent transportation system* framework can be extremely useful in preventing traffic accidents. Intelligent camera network installations in these settings can also be used to ease traffic congestion by keeping track of open spaces for slower drivers who are looking for parking spots, and to alert authorities when accidents occur. Finally, networks of smart cameras can aid in *automatic foul detection and commentary* during sporting events such as tennis and basketball, which can be very useful for fans who want specific information during game play.

1.1.1 Automated Perception System Requirements

It is clear from the several motivating scenarios presented previously that the first task of situation awareness, i.e. perception, needs to be robust to error and scalable. The three fundamental requirements for such an automated system are presented in what follows:

1. *The system should be composed of wireless smart cameras capable of performing basic image processing, and transmitting meta-data to a central server*

   There are several advantages to using wireless cameras instead of wired solutions. One of the primary advantages is the ease of installation. Wired cameras require significant resources and infrastructure for connecting each camera to the network and for supplying power to each one. Wireless cameras on the other hand, are easy to install, can be reconfigured to obtain any network topology, and existing networks can easily expand to incorporate new wireless camera nodes when it is desired. This versatility makes wireless solutions desirable in scenarios that require situational assessment.

In order to have a truly scalable camera network for situational assessment, each camera should possess some basic processing capabilities on board each node, while off-loading computationally intense operations to a central processing station that they are communicating with. One of the primary limitations of using wireless cameras is the large amount of energy required to transmit image streams to a central monitoring station. For example, a camera sensor that consumes 10 mW of power for simple image compression would require over 1 Watt of power to transmit each image frame to distances between 10-100 feet. Such high transmission power requirements can drain a single AAA battery in less than a month. Performing simple perceptual operations on-board the smart cameras and transmitting the extracted meta-data to a central processing station for further analysis, can significantly reduce power requirements for the wireless smart camera nodes.

2. *Data from all wireless smart cameras should be fused effectively at the central server*

   The process of partitioning the algorithmic operations has several advantages beyond saving power on smart camera nodes. For instance, the basic perceptual operations on board the wireless smart cameras can be carefully chosen so that they are largely stabilized while requiring minimal updates. Hence, only the central server would need to be upgraded when new semantic categories are included, or new algorithms for comprehension are developed. This versatility will also enable easy integration of more wireless smart camera nodes in the network, thereby providing scalability and robustness. Another significant advantage of global inference is the improved situation assessment accuracy that can be achieved by fusing information extracted from sensors placed at different vantage points.

3. *Data fusion algorithms should be capable of performing multiple computer vision tasks*

   From the motivating scenarios presented in the previous section, it is clear that the same network of smart cameras should be used for addressing multiple tasks, namely - detection, segmentation and categorization of objects, humans and their actions. Thus, it is necessary to design computer vision algorithms that treat all these sub-problems in unison. Further, designing such fusion frameworks can significantly improve the accuracy of the solutions to each individual sub-problem.

   For instance, object recognition and human detection are clearly related problems under realistic scenarios. This can be seen in the security and surveillance setting, where it is important to monitor unattended bags in any area. Recognizing an object as a bag and then associating it to a specific person can be challenging if the two problems are treated independently. If the two problems are tackled simultaneously however, then it becomes easier to narrow down the possible categories of an object that has been in contact with a person, thereby simplifying the semantic labeling process. Segmentation and categorization are also very related problems. Isolating the image regions corresponding to a person (or object) in an image can help identify who (or what) has been segmented. This is because segmentation would suppress the remaining parts of the image that could potentially confuse detectors. By the same token, using a learned model for a human or object can significantly help isolate

their corresponding regions in the image. Finally, gesture and activity recognition algorithms can significantly be improved if only the exact regions of the video corresponding to people are segmented and tracked over time and provided as input to these recognition algorithms. Further, recognizing what a person is doing before he completes the task can provide some form of motion model for tracking algorithms to predict where the person is going to be in successive frames.

1.2 Thesis Goals and Contributions

The goal of this thesis is to design an automated perception system for situation awareness. We have two main objectives - our preliminary objective is the development of a system that is capable of categorizing objects, landmarks and human actions in a distributed manner. In this regard, we are interested in designing a system that is composed of networks of smart cameras for local perception and a central station for global inference. Our secondary objective is to perform an analysis of the interplay between different computer vision tasks such as segmentation and categorization of objects. We focus on developing fusion frameworks that leverage the solutions of one task to improve the performance of the other. Specifically in this thesis, we propose the following contributions.

1.2.1 Object Categorization using Wireless Camera Networks

We begin this thesis by proposing an efficient distributed object categorization system for sensing, compression, and recognition of 3-D objects and landmarks using a network of wireless smart cameras. Our system is partitioned into distributed image feature extraction (performed on a wireless smart camera) and centralized inference (performed at a base station). While there are several approaches for object recognition, we focus on the Bag-of-Word (BoW) based models for representing objects and landmarks [20]. These histogram based models rely on the detection of image features that are scale invariant and to a large extent, view-point invariant. Such features are easy to associate across different vantage points, thereby reducing the complexity of matching query and target images of objects. The foundation of our work is based on the observation that histograms constructed from these image features exhibit a certain degree of sparsity. Motivated by the emerging theory of compressive sensing, we overview a sparsity based distributed sampling scheme to compress feature histograms that concisely represent the appearance of a common object observed by multiple cameras from different vantage points. We demonstrate that the corresponding feature histograms can be efficiently compressed in a distributed fashion, and the joint signals can be simultaneously decoded based on distributed compressive sensing theory.

First, we propose a new multiple view object database as a public platform to benchmark our system, which we referred to as the Berkeley Multiview Wireless (BMW) database. It captures the 3-D appearance of 20 landmark buildings sampled by five low-power, low-resolution camera sensors from multiple vantage points. We assume the camera sensors and the network station are connected only through a band-limited wireless channel. Then we review and benchmark state-

of-the-art methods to extract image features and compress their sparse representations. Finally, we propose a fast multiple-view recognition method to jointly classify the object observed by the cameras. We demonstrate that our multiple-view classification improves the performance of object recognition upon the traditional per-view classification algorithms in both small-baseline and large-baseline situations. Further, our system is capable of adapting to different network configurations and varying wireless bandwidth.

1.2.2 Segmentation of Informative Features for Improved Object Categorization

The accuracy of BoW classifiers is often limited by the presence of uninformative features extracted from the background or irrelevant image segments. Most existing solutions to prune out uninformative features rely on enforcing pairwise epipolar geometry via an expensive structure-from-motion (SfM) procedure. Such solutions are known to break down easily when the camera transformation is large or when the features are extracted from low-resolution, low-quality camera networks. In this thesis, we we explore the use of Sparse PCA as a variable selection tool for segmenting informative features in the object images captured from our low-resolution camera sensor networks. First, we show that using a large-scale multiple-view object database, informative features can be reliably identified from a high-dimensional visual dictionary by applying Sparse PCA on the histograms of each object category. Second, we propose a state-of-the-art algorithm that improves the speed of Sparse PCA using the Augmented Lagrange Multiplier (ALM) approach [66]. The new solver outperforms the state of the art for estimating sparse principal vectors as a basis for a low-dimensional subspace model. To mitigate the high dimensionality of the visual dictionary, a direct variable elimination method called SAFE is presented to prune out uninformative features for object recognition prior to the Sparse PCA process. We compare our implementation (SPCA-ALM) with existing algorithms on simulated data. The experiments shows that our algorithm outperforms previous convex programming approaches in terms of speed while maintaining the same estimation accuracy. Finally, we perform object recognition experiments on the BMW database, which demonstrate improved recognition by successfully suppressing uninformative features.

1.2.3 Human Activity Detection and Categorization

Automatic recognition of human actions in video has been a highly addressed problem in robotics and computer vision. Majority of the work in literature has focused on classifying pre-segmented video clips, and some progress has also been made on joint detection and recognition of actions in complex video sequences. These methods, however, are not designed for wireless camera networks where the sensors have limited internal processing and communication capabilities.

In this thesis, we extend our distributed recognition pipeline to the spatio-temporal setting for joint detection and categorization of human actions. The foundation of our work is based on Deformable Part Models (DPMs) for detecting objects in static images [32]. We have extended this framework to the single-view and multi-view video setting to jointly detect and recognize actions.

We call this the Deformable Keyframe Model (DKM) and tightly integrate it within a centralized video analysis system. Our system can handle video sequences captured by a single or multiple wireless smart cameras with overlapping views. Each wireless smart camera in our system is capable of extracting, encoding, and transmitting a feature vector corresponding to foreground objects of interest in every frame where motion is detected. At the base station, feature vectors from a single or multiple camera sources are fused within a graphical model framework for localizing and categorizing actions of interest. Our analysis demonstrates that this decoupling of the algorithm pipeline can significantly minimize the power and bandwidth consumed by the wireless cameras.

We experimentally validate our DKMs on two data sets. We first demonstrate the competitiveness of our algorithm by comparing its performance against other state-of-the-art methods, on a publicly available dataset. Then, we extensively validate our system on a novel dataset called the Bosch Multiview Complex Action (BMCA) dataset. Our dataset consists of 11 actions continuously performed by 20 different subjects while being captured by cameras located at 4 different vantage points. In our experiments, we demonstrate that the presence of multiple-views improves the performance of action detection and categorization significantly over the single-view setting.

1.2.4 Joint Frameworks for Segmentation and Categorization

One of the most important tasks for a situation awareness system is to detect people and objects of interest precisely in image streams. In many situations, objects can occur in environments where the background has similar color or texture when compared to some parts of the object. This introduces and extra level of complexity when the image pixels corresponding to the object needs to be segmented and categorized. Several formulations based on Random Fields (RFs) have been proposed for joint categorization and segmentation (JCaS) of objects in images [7, 11, 51]. The RF's sites correspond to pixels or superpixels of an image and one defines potential functions (typically over local neighborhoods) which define costs for the different possible assignments of labels to several different sites. Since the segmentation is unknown a priori, one cannot define potential functions over arbitrarily large neighborhoods as that may cross object boundaries. Categorization algorithms extract a set of interest points from the entire image and solve the categorization problem by optimizing cost functions that depend on the feature descriptors extracted from these interest points. There is some disconnect between segmentation algorithms which consider local neighborhoods and categorization algorithms which consider non-local neighborhoods.

In this thesis, we propose to bridge this gap by introducing a novel formulation which uses models of objects with deformable parts, classically used for object categorization, to solve the JCaS problem. We use these models to introduce two new classes of potential functions for JCaS; (a) the first class of potential functions encodes the model score for detecting an object as a function of its visible parts only, and (b) the second class of potential functions encodes *shape priors* for each visible part and is used to bias the segmentation of the pixels in the support region of the part, towards the foreground object label. We show that most existing deformable parts formulations can be used to define these potential functions and that the resulting potential functions can be optimized exactly using min-cut [ref]. As a result, these new potential functions can be integrated with most existing RF-based formulations for JCaS. We evaluate our JCaS algorithm on a publicly

available image database which consists of articulated full-body images of people. Our results clearly show the advantages of fusing segmentation and categorization into a joint framework.

1.3 Thesis Outline

The rest of the thesis is outlined as follows. In Chapter 2, we present our work on object and landmark categorization using static images captured by distributed smart cameras. We first present a review of existing object categorization algorithms with specific emphasis on Bag-of-Word and Deformable Part Model based frameworks. We then introduce our pipeline for distributed object and landmark recognition for wireless smart cameras, and demonstrate its performance on a challenging data set. In Chapter 3, we study how to improve Bag-of-Word based object categorization by suppressing uninformative background features. We explore the use of Sparse PCA as a variable selection tool for segmenting only informative features in the object images captures from wireless smart cameras, and demonstrate improved categorization by successfully suppressing uninformative features. In Chapter 4, we extend our distributed object categorization pipeline to the dynamic setting, and tackle the problem of detection and categorization of human actions in distributed smart camera networks. Our framework effectively fuses spatiotemporal information from multiple smart cameras using template co-occurrence constraints similar to DPMs. We perform several experiments on challenging data sets and demonstrate improved detection and categorization. In Chapter 5, we present a framework for joint segmentation and categorization of objects in images. Specifically, we propose a novel energy function that efficiently fuses DPM based energy functions with traditional energy functions used for binary segmentation of foreground pixels corresponding to an object of interest from the background. Finally, we summarize the conclusions of this work in Chapter 6 and discuss future avenues of research that are based on the ideas proposed in this thesis.

Chapter 2

Distributed Object Categorization

2.1 Introduction

In this chapter we discuss a distributed object categorization system using a network of wireless smart cameras. Distributed object categorization is a fast-growing research topic [98, 2, 112, 13, 12, 4], mainly motivated by the proliferation of portable camera devices and their integration with modern wireless sensor network technologies. Given a wireless network of cameras, this new paradigm studies how to classify a 3-D object that may be captured from multiple vantage points. This setup can be extremely useful for automated surveillance and security applications, especially in large areas with high foot-traffic. For instance, automatic recognition of objects such as unattended baggage in airports, or number of cars and pedestrians in a given city block, can be very useful for authorities for general monitoring, task planning and decision making. The ability to acquire multiple-view observations of a common object can effectively compensate many visual nuisances such as object occlusion and pose variation, and may further boost the recognition accuracy if the multiple-view images are properly utilized.

2.1.1 Literature Review

Recent studies in distributed object recognition can be summarized in four intimately related areas. The first area is focused on the development of smart camera platforms. In recent years, several experimental platforms have successfully integrated high-resolution cameras (together with other sensing modalities) with state-of-the-art mobile processors and considerable amounts of memory. The reader is referred to [73] for more details in this area.

The second area concerns the extraction of dominant image features to represent the 3-D objects that are captured in the images. Leveraging the available processing power of many smart cameras, these image features can be directly extracted on the camera sensor without relaying the full-resolution images to a base-station computer. Then, the choice of optimal object features for particular applications boils down to two factors: on one hand, the efficiency to compute these image features on the smart sensor; on the other hand, the accuracy to concisely represent the 2-D appearance of the objects.

There are different cues of information one could use from a training set of still images to learn models for object categories. Many approaches use appearance patches around salient points [17, 35, 60, 71], or patches using dense grid sampling on the training images [24, 27]. Shape is also an important cue for object categorization. Kumar et. al. [76] used object contours as shape features within a pictorial structure framework; Boundary curves and boundary fragments have also been used to represent the shape of many object classes [34, 72]. Jurie and Schmid [48] detect circular arc features from boundary curves and use these as salient points for matching. While all these approaches have been successful in restricted regimes, they have their own independent drawbacks. One of the primary drawbacks is due to the lack of viewpoint invariance of image patch and shape descriptors.

For these reasons, recent approaches based on local invariant feature detectors have become increasingly popular. The Scale Invariant Feature Transform (SIFT) introduced by Lowe [62] is one such feature descriptor that combines shape and appearance cues in a common feature descriptor. The success of SIFT as a viewpoint invariant feature detector has led to the development of other improved feature detectors and descriptors, such as SURF [39] and CHoG [97], which are better suited for deployment on mobile camera platforms. Typically, local features are first extracted independently in both a reference and a test image, then characterized by invariant descriptors and finally associated to each other based on a matching score. The success of these methods is mainly due to the viewpoint invariance of the feature descriptors, and the tolerance to clutter and occlusions.

The third area deals with the models for representing 3-D objects captured from multiple vantage points by static cameras. In one approach, feature descriptors form the basis for a codebook representation of object categories [5, 100, 60, 35, 16]. The codebook is in essence a learned selection of feature descriptors from a corpus of training images. A particular instantiation of an object class in an image is then composed form codebook entries, possibly arising from different source images. Different codebook methods differ in the feature descriptors used, and codebook learning methods. These methods are also referred to as ”Bag-of-Word” models as they do not consider geometric relations between parts[88, 20, 17]. The other common approach for 3-D object representation in static images deals with modeling the geometric relationship between feature templates. These approaches build on the pictorial structures framework [30, 33]. Pictorial structures represent objects by a collection of parts arranged in a deformable configuration. Each part captures local appearance properties of an object while the deformable configuration is characterized by spring-like connections between certain pairs of parts. The notion that objects can be modeled by parts in a deformable configuration provides an elegant framework for representing object categories. While these models are appealing from a conceptual stand-point, it is generally difficult to establish their value in practice. On difficult datasets, deformable models are often outperformed by conceptually weaker models such as bag-of-words.

The fourth area concerns the correspondence and compression of image features extracted from the multiple camera views. In a per-view basis, [13] argued that reliable feature correspondence can be established in a much lower-dimensional space between camera sensors, even if the feature vectors are linearly projected onto a random subspace. With multiple camera views, [12] studied a SIFT-feature selection algorithm, where the number of SIFT features that need to be transmitted

to the base station can be reduced by considering the joint distribution of the features among multiple camera views of a common object. A recent work [74] further considered using robust structure-from-motion techniques (e.g., RANSAC) to select strong object features between two camera views that satisfy an epipolar constraint induced by a large baseline transformation, and subsequently reject weak features as outliers from the final stage of object recognition.

2.2 Methods and Contributions

In this chapter we present a systematic study that focusses on distributed object recognition in low-power wireless smart camera networks. The work is based on an open-source smart camera platform, called CITRIC [73], which integrates a high-resolution camera with a 600 MHz fixed-point mobile processor and 80 MB memory. First, we propose a new multiple-view object database as a public platform to benchmark the system, which is referred to as the Berkeley Multiview Wireless (BMW) database.

We assume the camera sensors and the network station are connected only through a band-limited wireless channel. These assumptions impose stringent computation and network communication constraints on the system. Given such limited computation on-board the sensors and minimal network bandwidth, the bag-of-words model presents the most robust and flexible framework to represent objects and landmarks. Although bag-of-word representations compress visual information significantly, we observe that under extremely stringent communication constraints it would be beneficial to compress the visual information as much as possible. Motivated by the emerging theory of compressive sensing (CS), we overview a sparsity-based distributed sampling scheme to compress bag-of-word based visual histograms that concisely represent the appearance of a common object in multiple views [4]. We present the most recent developments in CS theory to effectively recover sparse signals using fast ℓ_1-minimization (ℓ_1-min) algorithms.

Finally, we propose a multiple-view recognition method to jointly classify objects observed by multiple cameras in the network, a concept that has been largely ignored by existing solutions. We show that the multiple-view classification significantly improves the performance upon traditional per-view classification algorithms in both small-baseline and large-baseline situations. Furthermore, the system is capable of adapting to the change in different network configurations and the wireless bandwidth.

2.3 Berkeley Multiview Wireless Database

In the literature, there exist several public image-based object recognition databases, such as Oxford Buildings [42], COIL-100 [81], and Caltech-101 [52]. However, most of the databases are constructed using high-resolution cameras that do not take into account the real-world noise and distortion exhibited by most low-power camera sensors in surveillance applications. In addition, some databases only capture object images in lab-controlled indoor environments (such as COIL-100), while others collect a wide variety of object images in the same categories that may not necessarily share the same appearance in 3-D (such as Caltech-101). To aid peer evaluation of

distributed object recognition methods for the wireless surveillance scenario, we have constructed a public multiple-view image database, namely, the BMW database. The database can be accessed online at: http://www.eecs.berkeley.edu/~yang/software/CITRIC/.

The BMW database consists of multiple-view images of 20 landmark buildings on the campus of University of California, Berkeley. For each building, 16 different vantage points have been selected to measure the 3-D appearance of the building. The apparatus for image acquisition incorporates five low-power CITRIC camera sensors [73] on a tripod, which can be triggered simultaneously. Figure 2.1 shows the configuration of the camera apparatus. Figure 2.2 shows some examples of the captured building images. The cameras on the periphery of the cross are named Cam 0, Cam 1, Cam 4, Cam 3 with a counter-clockwise naming convention, and the center camera is named Cam 2. Thus, the BMW database has a total of 960 images.

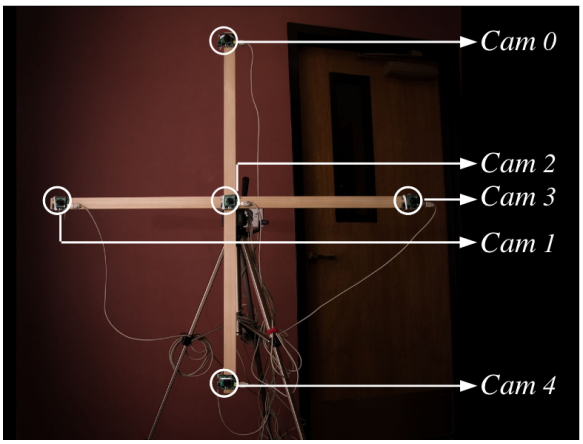
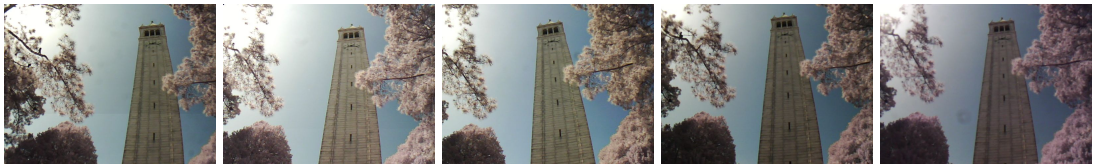
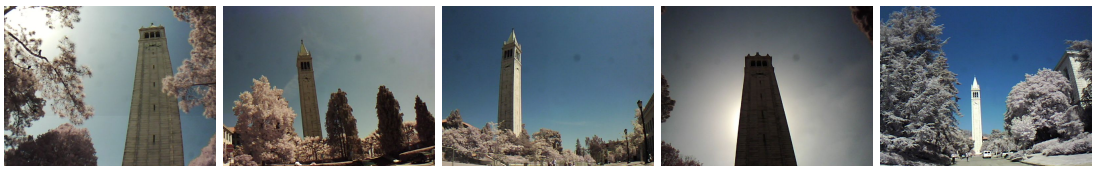


Figure 2.1: The apparatus that instruments five camera sensors.



(a) Five small-baseline images captured at one vantage point.



(b) Five large-baseline images captured at different vantage points.

Figure 2.2: Examples of multiple-view images of a building (the Campanile at UC Berkeley) in the BMW database.

It is worth emphasizing the following properties of the BMW database:

1. The images have been captured outdoor in different sessions. Therefore, some variations in ambient illumination exist within each building category and across different categories.

2. The image quality is considerably lower than many existing high-resolution databases, which is intended to reproduce realistic imaging conditions for camera surveillance applications. All images are 640×480 RGB color images. Since the CITRIC camera sensor does not have an auto-focus mechanism, the focal length of the camera is permanently set to maximum. However, it is noticeable that some images are slightly out of focus. In some cases, small image regions are visibly corrupted by dust residual on the camera lenses.

3. More importantly, the database provides a two-tier multiple-view relationship to systematically benchmark the performance of multiple-view object recognition algorithms, as shown in Figure 2.2. Specifically, the five images sampled at each vantage point simulate small-baseline camera transformations, while the images sampled at different vantage points simulate large-baseline camera transformations. Furthermore, the small-baseline image sets can be used to simulate the scenario where a slowly moving camera continuously sample images in a short time frame. In Section 2.6, we will systematically examine the recognition performance in both small-baseline and large-baseline scenarios.

2.4 Encoding Multiple-View Object Images via A Joint Sparsity Model

In this section, we briefly review a sparsity-based sampling scheme [4] to encode useful information in multiple-view object images from a distributed camera network. To implement a fast codec to recover distributed source signals in a sensor network setup, we also discuss the latest results on accelerated ℓ_1-min algorithms in the CS and optimization literature [3].

2.4.1 The Joint Sparsity Model

We assume multiple cameras are equipped to observe a common 3-D scene from different vantage points. For distributed object recognition, it is reasonable to simplify the communication model between sensors and the base station as a single-hop wireless network, i.e., the topology of the network is a star shape with the computer at the center and all the sensors directly communicate to the computer.

Using a SIFT-type feature detector, certain viewpoint-invariant features can be extracted from the images, as shown in Figure 2.3. For an object database (e.g., BMW), object features may be shared between different object classes. Therefore, all features extracted from the training images can be clustered/quantized based on their visual similarities into a vocabulary. The clustering normally is based on a hierarchical k-means algorithm [41]. The size of a vocabulary for a large database ranges from thousands to hundreds of thousands. For example, in this chapter, we use hierarchical k-means to construct 10,000-D vocabularies for the BMW database, with $k = 10$ and four hierarchies. Figure 2.4 shows the 10,000-D vocabulary tree constructed using all the CHoG features from the BMW training set (see Section 2.6 for more detail).

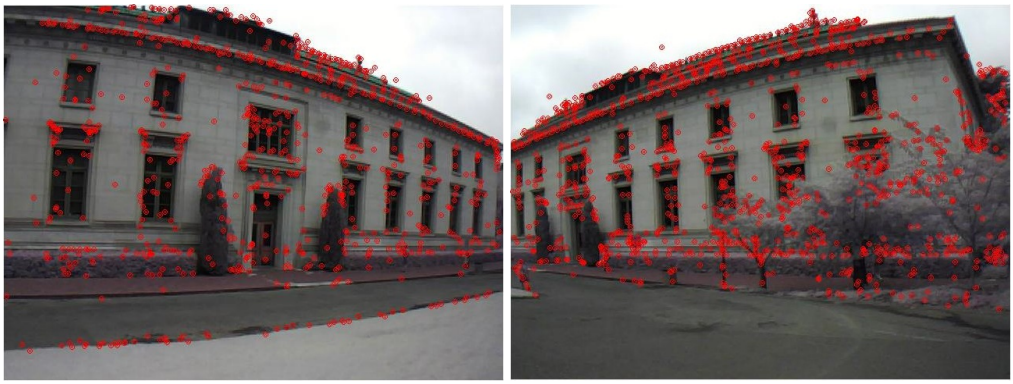


Figure 2.3: CHoG feature points detected in a pair of image views of a building.

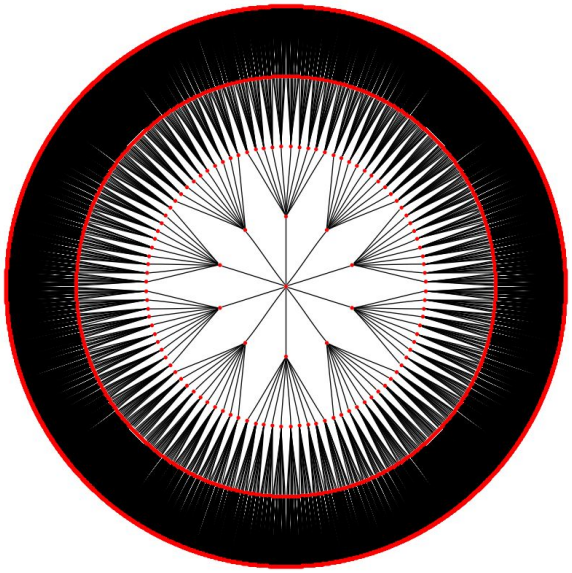


Figure 2.4: The 10,000-D vocabulary tree built using all CHoG features extracted from the training images in the BMW database. The tree is radially represented, with the center being the root node.

In [4], the authors have argued that, given a large vocabulary that contains quantized SIFT features from many classes, the representation of the features extracted from a single image is *sparse*, which is called a SIFT histogram. If we denote L as the number of the camera sensors that observe the same object in 3-D, and $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_L \in \mathbb{R}^D$ are the corresponding SIFT histogram vectors. Then each coefficient in $\boldsymbol{x}_i$ represents the instances of one type of the SIFT feature in the i-th view. Since only a small number of the features may be exhibited in a single image, the majority of the histogram coefficients should be (close to) zero. More importantly, since SIFT-type features are robust to some degree of camera rotation and translation, images from different vantage points may share a subset of the same features, thus yielding histograms with similar coefficient values, as shown in Figure 2.5.

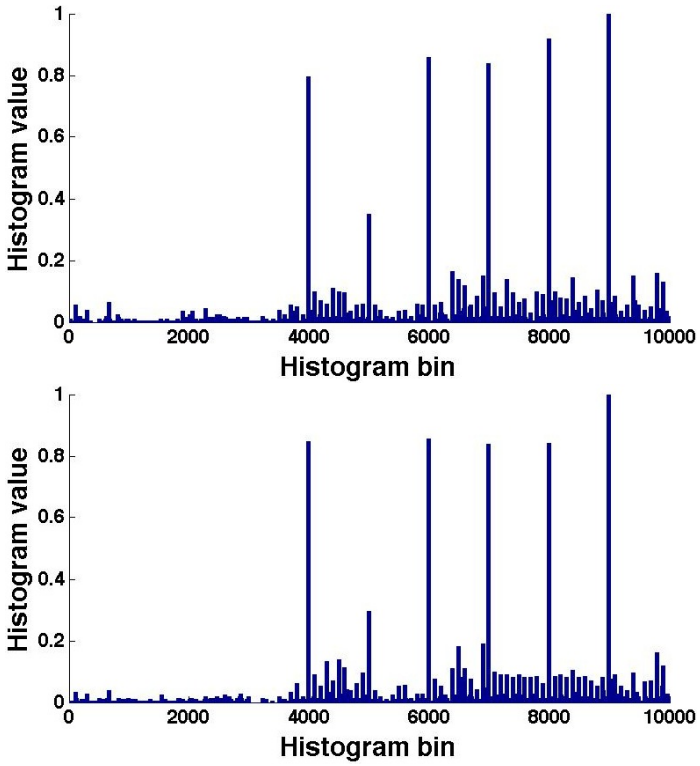


Figure 2.5: The 10,000-D feature histograms corresponding to the image pair in Figure 2.3. The joint sparsity patten indicates certain dominant features are shared between the two views.

The problem of encoding multiple-view object images can be formulated as the following. For the high-dimensional histogram vectors extracted from the L images, define a *joint sparsity* (JS) model as

$$\begin{aligned} \boldsymbol{x}_1 &= \boldsymbol{x}_c + \boldsymbol{z}_1, \\ &\vdots \\ \boldsymbol{x}_L &= \boldsymbol{x}_c + \boldsymbol{z}_L, \end{aligned} \tag{2.1}$$

where $\boldsymbol{x}_c$ represents the *common* component, and each $\boldsymbol{z}_i$ represents an *innovation*. Furthermore, both $\boldsymbol{x}_c$ and $\boldsymbol{z}_i$ are also sparse. On each camera sensor, an encoding function $\boldsymbol{b}_i \doteq f(\boldsymbol{x}_i) \in \mathbb{R}^d$ is sought to compress the histogram vector $\boldsymbol{x}_i$. At the base station, upon receiving $\boldsymbol{b}_1, \cdots, \boldsymbol{b}_L$ compressed features, the system should simultaneously recover the source signals $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_L$, and further proceed to classify the 3-D object represented by the multiple-view histograms.

2.4.2 Distributed Encoding of JS Signals

The fact that each $\boldsymbol{x}_i$ is sparse against a large vocabulary provides a means to effectively sample the signal via a linear projection, motivated by the CS theory. In particular, we define a *random* matrix $A \in \mathbb{R}^{d \times D}$ as an overcomplete dictionary (i.e., $d < D$) whose elements are sampled from

independent and identically-distributed Gaussians. Then a random projection function is defined as:

$$f : \boldsymbol{b} = A\boldsymbol{x}. \tag{2.2}$$

However, recovering $\boldsymbol{x}$ from (2.2) essentially is an inverse problem, as the number of observations in $\boldsymbol{b}$ is smaller than the number of unknowns in $\boldsymbol{x}$. The CS theory [14, 19] shows that if the underlying signal $\boldsymbol{x}_i$ is sufficiently sparse and the projection dimension $d > \delta(A)D$ is above a threshold determined by $\delta(A)$, then $\boldsymbol{x}_i$ is the unique solution to a convex program called ℓ_1-min:

$$(P_1): \quad \min \|\boldsymbol{x}\|_1 \text{ subject to } \boldsymbol{b} = A\boldsymbol{x}. \tag{2.3}$$

In other words, (P_1) guarantees that no information is lost by projecting $\boldsymbol{x}_i$ onto a low-dimensional random subspace, as long as $\boldsymbol{x}_i$ is sufficiently sparse.

Now we can consider the decoding problem at the base station. Given the fact that all camera views may share a sparse component $\boldsymbol{x}_c$, the ensemble $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_L$ can be simultaneously recovered at the base station with the accuracy that may exceed that by estimating (P_1) individually [4]. In particular, the JS model can be solved in a single linear system:

$$\begin{aligned} \begin{bmatrix} \boldsymbol{b}_1 \\ \vdots \\ \boldsymbol{b}_L \end{bmatrix} &= \begin{bmatrix} A_1 & A_1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \\ A_L & 0 & \cdots & 0 & A_L \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_c \\ \boldsymbol{z}_1 \\ \vdots \\ \boldsymbol{z}_L \end{bmatrix} \\ \Leftrightarrow \qquad \boldsymbol{b}' &= A'\boldsymbol{x}' \in \mathbb{R}^{dL}. \end{aligned} \tag{2.4}$$

Enforcing the JS model can boost the estimation accuracy in (P_1) when $d_1 = d_2 = \cdots = d_L = d$ is uniform. More importantly, it also makes it possible to choose different sampling rates for individual camera sensors. This property is particularly relevant to wireless sensor networks, where sensor nodes that have lower bandwidth or lower power reserve may choose to reduce their sampling rates in order to preserve energy.

More specifically, the strategy of choosing varying sampling rates can be viewed as an application of the celebrated Slepian-Wolf theorem [21]. For the simplest case of two source channels X_1 and X_2, the theorem shows that, given sequences x_1 and x_2 that are generated from the two channels respectively, the sequences can be jointly recovered with vanishing error probability asymptotically *if and only if*

$$\begin{aligned} R_1 &> H(X_1|X_2), \\ R_2 &> H(X_2|X_1), \\ R_1 + R_2 &> H(X_1, X_2), \end{aligned}$$

where R is the bit rate function, $H(X_i|X_j)$ is the conditional entropy for X_i given X_j, and $H(X_i, X_j)$ is the joint entropy.

In distributed object recognition, with the JS model, a *necessary* condition for simultaneously recovering $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_L$ can be found in [18]. Basically, it requires that each sampling rate $\delta_i = \frac{d_i}{D}$ guarantees the so-called *minimal sparsity signal* of $\boldsymbol{z}_i$ is sufficiently encoded, and also the total sampling rate guarantees that both the joint sparsity and the innovations are sufficiently encoded.

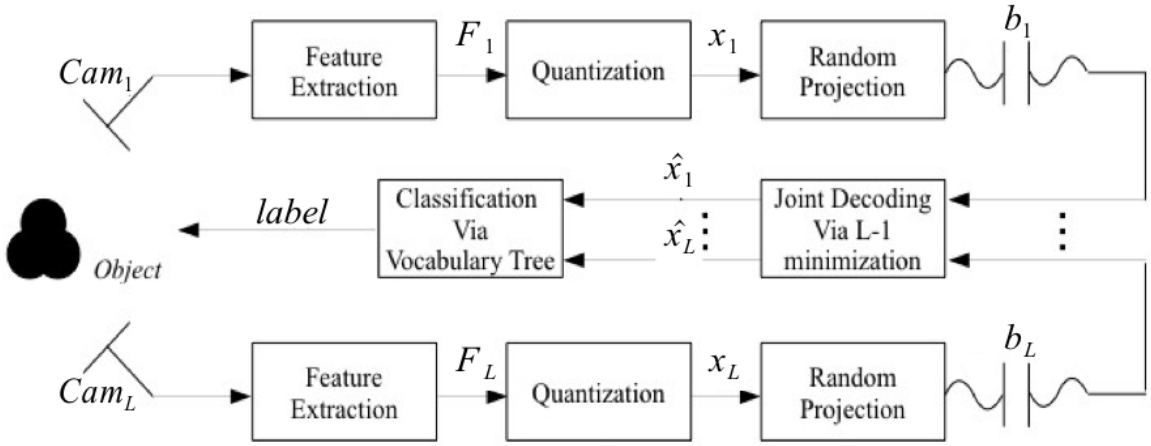


Figure 2.6: Flow diagram of the sparsity-based distributed object recognition system.

2.4.3 Decoding Sparse Signals via Fast ℓ_1-Minimization Algorithms

Finally, we briefly discuss the state of the art in effectively solving the convex program (P_1) via an accelerated ℓ_1-min technique. A comprehensive review of existing fast ℓ_1-min algorithms can be found in [3].

The convex program (P_1) has traditionally been formulated as a linear programming problem called *basis pursuit* (BP), which has several well-known solutions via iterative interior-point methods. However, the computational complexity of these interior-point methods is often too high for many real-world, large-scale applications. The main reason is that they all involve expensive operations such as matrix factorization and solving linear least squares.

Recently, *iteraive shrinkage-thresholding* (IST) methods have been recognized as a good alternative to the exact BP solutions. The approach is also appealing to large-scale applications because its implementation mainly involves lightweight operations such as vector operations and matrix-vector multiplications, which is in contrast to most past ℓ_1-min algorithms.

In a nutshell, IST considers a variation of (P_1) that takes into account the existence of measurement errors in the sensing process:

$$(P_{1,2}): \quad \min \|\boldsymbol{x}\|_1 \text{ subject to } \|\boldsymbol{b} - A\boldsymbol{x}\|_2 \leq \epsilon, \tag{2.5}$$

where ϵ is a bound on the additive white noise in $\boldsymbol{b}$. By the Lagrangian method, $(P_{1,2})$ is rewritten as an unconstrained *composite objective function*:

$$\min_{\boldsymbol{x}} F(\boldsymbol{x}) \doteq \frac{1}{2}\|\boldsymbol{b} - A\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1, \tag{2.6}$$

where $\lambda > 0$ is the Lagrangian multiplier.

We can immediately see that the main issue in optimizing such a composite function $F(\boldsymbol{x})$ is that its second term $\|\boldsymbol{x}\|_1$ is not a smooth function and therefore is not differentiable everywhere.

Nevertheless, one can always locally linearize the objective function in an iterative fashion as [82, 1]:

$$\begin{aligned} \boldsymbol{x}^{(k+1)} &= \arg\min_{\boldsymbol{x}}\{f(\boldsymbol{x}^{(k)}) + (\boldsymbol{x}-\boldsymbol{x}^{(k)})^T\nabla f(\boldsymbol{x}^{(k)}) \\ &\quad +\tfrac{1}{2}\|\boldsymbol{x}-\boldsymbol{x}^{(k)}\|_2^2 \cdot \nabla^2 f(\boldsymbol{x}^{(k)}) + \lambda g(\boldsymbol{x})\} \\ &\approx \arg\min_{\boldsymbol{x}}\{(\boldsymbol{x}-\boldsymbol{x}^{(k)})^T\nabla f(\boldsymbol{x}^{(k)}) \\ &\quad +\tfrac{\alpha^{(k)}}{2}\|\boldsymbol{x}-\boldsymbol{x}^{(k)}\|_2^2 + \lambda g(\boldsymbol{x})\}, \end{aligned} \tag{2.7}$$

where the hessian $\nabla^2 f(\boldsymbol{x}^{(k)})$ is approximated by a diagonal matrix $\alpha^{(k)}I$.

One can further show that the linearized objective function (2.7) has a closed-form solution called the *soft-thresholding* function [82, 1]. Furthermore, the speed of convergence from an initial guess $\boldsymbol{x}^{(0)}$ to the ground-truth sparse signal can be *accelerated* by a numerical technique called the *alternating direction method* (ADM) [45]. Based on the IST algorithm, ADM iteratively optimizes both the sparse signal $\boldsymbol{x}$ and the residual term $\boldsymbol{e}$:

$$\min_{\boldsymbol{x},\boldsymbol{e}} \|\boldsymbol{x}\|_1 + \frac{1}{2\mu}\|\boldsymbol{e}\|^2 \text{ subject to } \boldsymbol{b} = A\boldsymbol{x} + \boldsymbol{e}. \tag{2.8}$$

It is easy to see that when $\boldsymbol{e}$ is fixed, (2.8) can be converted to the standard IST problem in (2.7); when $\boldsymbol{x}$ is fixed, since the ℓ_1-norm $\|\boldsymbol{x}\|_1$ becomes a constant, the objective function becomes smooth and its optimum is trivial to compute.

2.5 Multiple-View Object Recognition using a Hierarchical Vocabulary Tree

In this section, we explain an efficient multiple-view object recognition algorithm that takes multiple-view histograms as the input, and outputs a label as the classification of the object in 3-D. Figure 2.6 summarizes the complete system diagram.

Given a large set of robust image features (e.g., SIFT), we can construct a vocabulary tree using hierarchical k-means, where k represents the branch factor of the tree [20]. On the highest level of the tree, all the feature descriptors are partitioned into k clusters, with the mean of each cluster representing the cluster center. At each lower level, k-means is applied within each previous cluster in order to further partition the space into k clusters. The process is continued until there are k^H clusters at the H-th level (as shown in Figure 2.4).

With the vocabulary tree constructed, the feature descriptors in each training image are propagated down the tree. Then a term-frequency inverse-document-frequency (*tf-idf*) weighted histogram $\boldsymbol{y}$ can be defined for each training image as follows. First, assign an entropy-based weight w_p to each quantized leaf node feature p in the vocabulary tree as

$$w_p \doteq \log\frac{N}{N_p}, \tag{2.9}$$

where N is the total number of the training images, and N_p is the number of training images that contain the same feature vector p. With the weight w_p computed in this manner, all the elements of

the training histograms $\boldsymbol{y}$ and test histograms $\boldsymbol{x}$ are multiplied element-wise with this weight function in order to achieve the *tf-idf* weighting scheme. For each object category, $i = 1 \cdots C$, multiple weighted histograms are generated for all m training images of that object and grouped into a set, $Y_i = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_m\}$. All the C sets further form the training set, $Y = \{Y_1, Y_2, \cdots, Y_C\}$.

During the testing phase, feature descriptors are extracted for each single-view query image and propagated down the vocabulary tree by the same fashion to obtain a single weighted query histogram $X = \{\boldsymbol{x}\}$. The query image is then given a single-view relevance score s based on the ℓ_1-normalized difference between the weighted query and the ith training set Y_i:

$$s(x, Y_i) = \min_{\boldsymbol{y}_j \in Y_i} \| \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_1} - \frac{\boldsymbol{y}_j}{\|\boldsymbol{y}_j\|_1} \|_1. \tag{2.10}$$

When multiple-view histograms of the query object are available, $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_L\}$, a new method to perform joint classification is necessary to take into account the multiple-view information. In this case, the median of the single-view relevance scores is used to determine the averaged multiple-view relevance score:

$$s(X, Y_i) = \text{median}_{\boldsymbol{x}_j \in X} s(\boldsymbol{x}_j, Y_i). \tag{2.11}$$

We choose median as a robust mean operator, which is more suitable for situations where some query images are not well matched with any training images in 3-D. Notice that when X only contains a single camera view, (2.11) is identified as (3.1).

Finally, the label of the object category for the multiple-view histograms is assigned as:

$$\text{label}(X, Y) = \arg \min_{i \in [1 \cdots C]} s(X, Y_i), \tag{2.12}$$

which is simply the object category that achieves the minimal multiple-view relevance score.

In this chapter, we are concerned with the implementation of the above multiple-view recognition system on a band-limited camera sensor network. As shown in Figure 2.6, on the sensor side, each query histogram after the quantization process is projected onto a lower-dimensional feature space by random projection, and transferred to a base-station computer. On the computer side, the received feature vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_L$ are jointly decoded in (2.4) by ℓ_1-min to obtain the estimates $X = \{\hat{\boldsymbol{x}}_1, \hat{\boldsymbol{x}}_2, \cdots, \hat{\boldsymbol{x}}_L\}$ of the original weighted histograms. Finally, the joint classification algorithm (3.2) is employed to recover a label of the object that minimizes the multiple-view relevance score s.

2.6 Experiments

2.6.1 Setup

We use the BMW database to benchmark the performance of the algorithm (3.2). First, we divide the database into a training set and a testing set. As the vantage points of each object are named numerically from 0 to 15, images from all the even number locations are designated as the training

set, and the ones from the odd number locations are assigned to the testing set. Furthermore, since the main purpose of the experiment is to validate the recognition performance of using multiple-view *testing* images, we do not include the redundant multiple views in the training set. More specifically, only training images from a single camera, i.e., Cam 2, are used for the construction of the vocabulary tree and for the subsequent recognition process.

Based on the BMW database, we choose to compare how discriminative three existing robust feature descriptors are in representing the image appearance of objects, namely, SIFT[62], SURF[39], and CHoG[97]. The original SIFT framework includes a gradient-based interest-point detector with a single-scale 128-D descriptor for each feature. The SURF algorithm is based on sums of approximated 2D Haar wavelet responses, and it also makes use of integral images to speed up the keypoint detection and descriptor extraction. The quantization process yields a 64-D vector. The relatively newer CHoG feature detector and descriptor has been specially designed for platforms with low processing capabilities, and yields a 45-D descriptor for each detected feature.

We design two testing scenarios to evaluate the performance of the distributed recognition scheme, namely, the small-baseline and the large-baseline scenarios. In the small-baseline scenario, images captured concurrently from multiple cameras at one vantage point are used to determine the object category. We evaluate the recognition performance using one camera (i.e., Cam 2), two cameras (i.e., Cam 1 and Cam 2), and three cameras (i.e., Cam 1, Cam 2, and Cam 3). In the large-baseline scenario, images captured from one to three vantage points are randomly chosen from the same testing category for recognition. The two scenarios are well illustrated in Figure 2.2.

In terms of system implementation, the CITRIC mote has been shown to have the capacities to locally extract and compress high-dimensional histograms [4]. Nevertheless, in this chapter, the data processing and classification on the BMW database are performed on a Linux workstation. All the code has been implemented in MATLAB/C++ with a MEX compiler interface.

2.6.2 Small-Baseline Results

To establish a baseline performance, we first evaluate the recognition accuracy of (3.2) without involving the random-projection and ℓ_1-min codec. In other words, we assume the classifier can directly access and process all the images in their full resolution. Table 2.1 shows the recognition rates for the three camera configurations based on the SIFT, SURF, and CHoG feature descriptors. It shows that in all the three cases, the recognition rates improve when more views of the query object are included in the global recognition scheme. Overall, CHoG features yield the best recognition rates compared to the other two feature descriptors. We find this to our benefit, as CHoG features have been designed for distributed wireless camera applications [97], and thus have the lowest dimensionality and extraction time compared to SURF and SIFT feature descriptors. For this reason, we will choose the CHoG features exclusively for the multiple-view recognition experiment in the rest of the section.

Next, we activate the ℓ_1-min codec in the same camera configurations, and evaluate the recognition accuracy when the query histograms are projected from its original 10,000-D space to lower projection dimensions ranging from 1000 to 9000. For each projection dimension d and each cam-

Table 2.1: Small-baseline recognition rates without histogram compression. The best rates are marked in bold face.

Expt.	# Train Images	# Test Images	SIFT Rate(%)	SURF Rate(%)	CHoG Rate(%)
1 Cam	160	160	71.25	80.62	**81.88**
2 Cam	160	320	72.5	81.25	**84.38**
3 Cam	160	480	73.75	81.88	**86.25**

era sensor j, we create a fixed random projection matrix A_{dj} offline. The ℓ_1-min algorithm to reconstruct the JS signals (2.4) is based on the alternating direction method [45]. Figure 2.7 shows the recognition rates for the three experiments against the projected dimension.

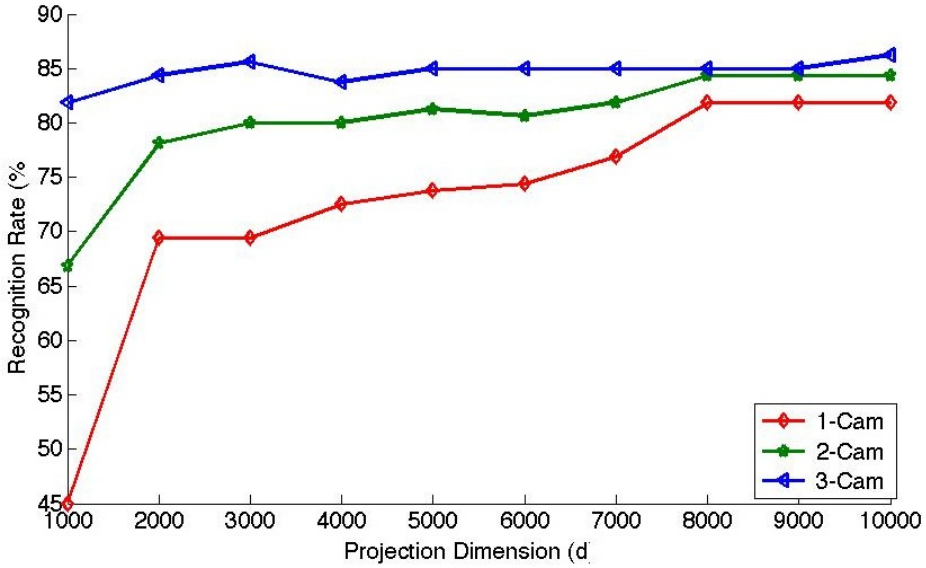


Figure 2.7: Comparison of the CHOG recognition rates (in color) in the small-baseline scenario with different random projection dimensions.

We observe that, with small projection dimensions close to 1000, the recognition rates using two or three cameras improves significantly compared to the single-view recognition rates. For instance, at $d = 1000$, the recognition rate from a single camera (i.e., Cam 2) is about 45%. The rate is boosted to 68% with two cameras and 82% with three cameras. It is also important to note that the improved recognition rates using the multiple-view information are also higher than merely increasing the projection dimension in the single-camera scenario. For instance, The recognition rate for 2-Cam at $d = 2000$ is higher than the rate for 1-Cam at $d = 4000$.

As the projection dimension increases, the recognition rates for the three scenarios increase as well and reach a plateau beyond $d = 8000$. Interestingly, for the 3-Cam case, the ground-truth recognition rate of 85% is achieved in a very low projection space of 3000-D.

2.6.3 Large-Baseline Results

The large-baseline performance is evaluated using the same procedure as in the small-baseline experiments. Table 2.2 shows the recognition rates for the three camera configurations without involving the ℓ_1-min codec. Again, the recognition rates improve when more views of the query object are included in the global recognition scheme. Recognition using the CHOG features not only outperforms that with the other two feature descriptors, but is also drastically better than the CHOG recognition rates in the small-baseline experiments of Section 2.6.2. Specifically, there is about 10% improvement in the recognition rates in the 3-camera case. The result demonstrates that multiple large-baseline images contain much more information about a common object in 3-D than a set of small-baseline images.

Table 2.2: Large-baseline recognition rates without histogram compression. The best rates are marked in bold face.

Expt.	# Train Images	# Test Images	SIFT Rate(%)	SURF Rate(%)	CHoG Rate(%)
1 Cam	160	160	71.25	80.62	**81.88**
2 Cam	160	320	76.88	88.13	**93.75**
3 Cam	160	480	83.13	90.00	**94.88**

When the ℓ_1-min codec is included, Fig. 2.8 shows the recognition rates versus the random projection dimension. Clearly, the recognition rates using a single camera does not change from the small-baseline scenario. As shown in the plot, the recognition rates at the low projection dimension of 1000 are lower than those of the small-baseline scenario for the 2 and 3-cam cases. However, as the projection dimension increases, the multiple-view recognition rates reach about 95% and begin to plateau. Such rates are never achieved even without random projection in the single view case.

2.7 Conclusion and Discussion

We have presented a framework to jointly classify objects observed from multiple vantage points in a distributed wireless camera network. The method is well suited for situations where the camera sensors and the base station are connected only by a band-limited communication channel, and the multiple-view information of the object is available to boost the global recognition. We have drawn from recent developments in compressive sensing theory to formulate a distributed compression scheme to transmit high-dimensional object histograms from camera sensors viewing a common object in 3-D. Most importantly, the algorithm does not require any calibration between the cameras. Therefore, it is very flexible to the addition or omission of some cameras in the network, and the cameras can also be mounted on mobile robot platforms. Finally, we have constructed a new multiple-view object database, namely, the BMW database. The performance of the system has been extensively validated using the database.

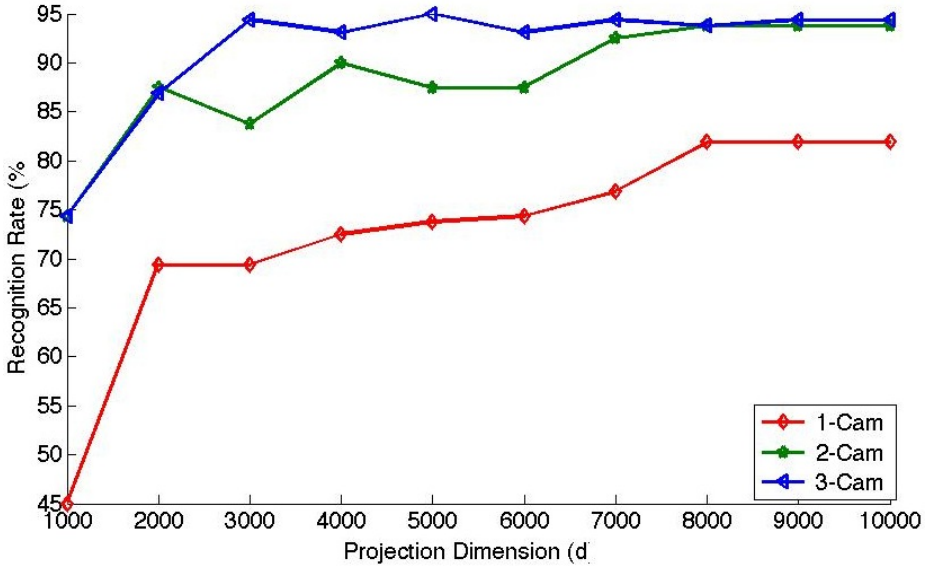


Figure 2.8: Comparison of the CHOG recognition rates (in color) in the large-baseline scenario with different random projection dimensions.

Our investigation also has led to several intriguing problems for further investigation. First, the multiple-view images may adversely introduce large amounts of outlying features from different background images into the recognition process. However, it is possible to reject these features by considering the geometric consistency between the multiple views during the (offline) training process, such as using the RANSAC technique in [74]. Second, the best recognition rate based on the images of the 20 landmarks is about 95%. To successfully deploy such systems in real-world surveillance applications, the recognition rates have to be improved dramatically (e.g., $>$ 99%). Finally, robust techniques must be studied to deal with situations where multiple objects of interest are captured in the images, or certain test images are irrelevant (as outliers) to the given training categories. In the next chapter, we address the first issue of segmenting informative features in multi-view images of a common object or landmark, and integrate it into our multi-view recognition framework.

Chapter 3

Segmenting Informative Features for Categorization via Sparse PCA

3.1 Introduction

In the past decade, the exponential growth of storage capacity and the proliferation of modern smartphones equipped with mobile cameras has enabled people to capture and upload personal images to large online image databases such as Picassa and Flickr. Further, the mobility and ease of use of smart phones has empowered users to capture images of objects and events in public spaces. This has enabled a host of applications that can automatically recognize common objects and landmark buildings in man-made urban environments. These applications range from location based services to augmented reality and computational photography. Further, these "crowdsourced" images and videos are also being used by news media and law enforcement for situation awareness in public spaces as they offer scene imagery from varying vantage points. These images in essence can be viewed as being captured form a virtual smart-camera network, and thus can be processed using the computational machinery that has been developed for multi-view camera networks.

The existence of common objects and landmarks in these images has motivated research in visual object recognition [59, 29, 46, 114]. Images in these coarsely labelled databases are used to train classifiers that can be used to recognize different object categories. To tackle the problem of recognizing a large number of objects in large image databases, a visual-dictionary based approach has been proposed [44, 20], which further led to several other methods to recognize objects in both the single-view and multi-view settings [98, 2, 112, 12, 4, 67]. Essentially, most of the methods work with certain visual descriptors (*e.g.*, SIFT and its many variants) extracted from the images to construct visual histograms, which represent the object appearance in the images using a precomputed visual dictionary.

Although vocabulary-tree methods have proven to be efficient in describing object images, the accuracy of the classifiers is often limited by the presence of uninformative image features typically extracted from the background or irrelevant image segments, such as pedestrians and vegetation (see Figure 3.1 for an example). When the irrelevant segments take on a significant portion of an image, the uninformative features can dominate the representation in the visual histogram, and

hence lead to inferior recognition accuracy. In [74], Turcot and Lowe suggested, if a subset of so-called *useful features* or *informative features* can be systematically selected during the training stage, it not only further reduces the number of visual descriptors needed, but also significantly improves the recognition accuracy. Since in man-made environments, most objects of interest, in particular landmark buildings, are rigid objects, 3-D perspective geometry can be leveraged to select informative features that satisfy a pairwise epipolar constraint via RANSAC. This is known as the Structure-from-Motion (SfM) approach.

Motivated by the literature, in this chapter, we study how to improve informative feature selection in both speed and accuracy from possibly *low-resolution*, *low-quality* camera networks. The proposed approach can be also applied to improve object recognition in the traditional sense using high-end photography [44, 20, 74]. One major difficulty in enforcing the epipolar constraint on images collected from low-power camera networks instead of high-end photography is that establishing wide-baseline feature correspondence of SIFT-type features is known to be not robust even when using state-of-the-art bundle adjustment techniques [90]. In addition, the quality of images sampled from low-power camera sensors also presents a challenge to reliably extract accurate features to describe the appearance of interesting objects in multiple views.

We propose to address this problem by a principled semidefinite programming (SDP) technique, known as Sparse Principal Components Analysis (Sparse PCA) [117]. As an extension of the popular PCA method, Sparse PCA addresses a drawback of classical PCA that the principal vectors (PVs) as a basis of a low-dimensional subspace typically have dense non-zero loadings. In particularly, in high-dimensionality setting, the dense linear combinations of all the variables make it difficult to interpret the corresponding principal components (PCs).

In case of visual-dictionary based object recognition, the variables in a high-dimensional histogram are associated with the codewords that represent either informative foreground features or uninformative background. We contend that in a large-scale object image database, the subset of informative features can be reliably selected by the sparse coefficients in the first few PVs. The new solution is more robust to wide-baseline camera transformation and numerically more efficient than the existing solutions of establishing pairwise rigid-body correspondence.

3.1.1 Main Contributions

In this chapter, we explore the use of Sparse PCA as a variable selection tool for selecting informative features in the object images captured from low-resolution camera sensor networks. Firstly, we present a scheme for using Sparse PCA with high-dimensional covariance matrices constructed from visual histograms to extract a sparse visual codeword support for each object category. We compare its performance with the SfM technique applied to large-baseline, low-quality multiple-view images. Secondly, we propose a state-of-the-art algorithm that improves the speed of Sparse PCA using the Augmented Lagrange Multiplier (ALM) approach [8, 3]. To mitigate the high dimensionality of the visual dictionary, a direct variable elimination method called SAFE is presented to prune out uninformative features for object recognition prior to the Sparse PCA process. We compare our implementation (SPCA-ALM) with existing algorithms on simulated data. The experiment shows that the algorithm outperforms the previous convex programming algorithm

Figure 3.1: Comparison of informative feature selection on low-quality multiple-view images. **Top:** A subset of 16 training images of a building (Campanile at UC Berkeley) in the BMW database [67] with SURF features superimposed in blue. **Middle-top:** Informative features detected by SfM (green). For each image pair, SURF features are deemed informative if the consensus of the corresponding epipolar constraint exceeds 25% of the total feature pairs. **Middle-bottom:** Informative features selected by thresholded PCA (pink), with desired cardinality equal to that of Sparse PCA **Bottom:** Informative features selected by Sparse PCA (red) based on the first two leading PVs. The selected features primarily lie on the Campanile, while other features on the trees, lamps, and other objects are successfully suppressed. For this particular dataset, the SfM method performs poorly due to unreliable epipolar transformations found between these wide-baseline images.

(DSPCA) [23] in terms of speed while maintaining the same estimation accuracy. Finally, we perform object recognition experiments, which demonstrate improved recognition by successfully suppressing uninformative features.

3.2 Review of Recognition via Vocabulary Trees

In object recognition, certain local invariant features have become a popular representation of the object images, which can be extracted and encoded into high-dimensional descriptors using algorithms such as SIFT [62] and SURF [39]. In bag-of-words (BoW) approach, these invariant features are further quantized to form a dictionary of *visual words*. All the feature descriptors in the training set are hierarchically clustered into visual word clusters (e.g., using hierarchical *k*-means [41]). This hierarchical tree is commonly referred to as a *vocabulary tree* [20]. The size of a vocabulary tree for a large database ranges from thousands to hundreds of thousands. For example, we use hierarchical *k*-means to construct 1,000-D vocabularies for our training image database, with a branch factor of $k = 10$ and four hierarchies.

To start the training process, feature descriptors in each training image are propagated down the vocabulary tree to form a BoW model for the image. Then a term-frequency inverse-document-frequency (*tf-idf*) weighted visual histogram $\boldsymbol{y}$ is defined for each training image [20]. For each object category, $i = 1 \cdots C$, m weighted histogram are generated from the m training images of that category respectively: $A_i = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_m\}$. All the C sets form the training set, $A = \{A_1, A_2, \cdots, A_C\}$.

During the testing phase, feature descriptors are extracted for the query image and propagated down the vocabulary tree by the same fashion to obtain a single weighted query histogram $\boldsymbol{q}$. The query image is then given a relevance score s based on the ℓ_1-normalized difference between the weighted query and the ith training set A_i:

$$s(\boldsymbol{q}, A_i) = \min_{\boldsymbol{y}_j \in A_i} \| \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|_1} - \frac{\boldsymbol{y}_j}{\|\boldsymbol{y}_j\|_1} \|_1. \tag{3.1}$$

Finally, the label of the visual histogram $\boldsymbol{q}$ is assigned as the object category that achieves the minimal relevance score:

$$\text{label}(\boldsymbol{q}) = \arg\min_{i \in [1 \cdots C]} s(\boldsymbol{q}, A_i). \tag{3.2}$$

3.2.1 Failure of SfM on low quality images

It was suggested by Turcot and Lowe [74] that the accuracy of object recognition in large image databases can be improved by suppressing uninformative visual words that typically represent irrelevant image background. In [74], SfM techniques were used to enforce pairwise epipolar constraints of rigid objects. The authors argued that, between a pair of images that render the same object in space, uninformative features can be easily pruned out as outliers w.r.t. a dominant epipolar constraint by RANSAC. Along similar lines, Philbin *et. al.* [43] introduced a Geometric Latent Dirichlet Allocation model for constructing image adjacency graphs. Subsequently, rich latent topic models were built from the adjacency graphs with the identity and locations of visual words specific to the objects, thereby rejecting uninformative visual words. Knopp *et. al.* [49] augmented query images with rough geolocation information combined with wide-baseline feature matching to detect and suppress uninformative features before invoking vocabulary tree based object recognition.

All these methods rely on the accuracy of wide baseline feature matching to establish pairwise epipolar geometry. However, they tend to fail when the quality of the images in the database is very poor, as is the case with images captured from mobile cellphones or distributed camera networks. Furthermore, man-made landmarks such as buildings often have repetitive texture and patterns that tend to confuse feature correspondence algorithms (*e.g.*, Bundler [90]). Figure 3.1 (Middle-top) shows an example where SfM fails at determining the wide-baseline transformation across images of an object captured from multiple vantage points. More examples can be found in Figure 3.4 later.

3.3 Identifying Informative Features

Classical PCA is a well established tool for the analysis of high-dimensional data. For a data matrix A, PCA computes the PCs via an eigenvalue decomposition of its empirical covariance matrix Σ. It has also been observed that in general the loadings of the corresponding PVs have dense and nonzero. In certain applications, it is desirable to obtain PVs that can explain maximum variability in the data A using linear combinations of just a few nonzero variables, and hence improves interpretability of such data. It is with this motivation that Sparse PCA was developed [117, 23] and has proven to be a very useful tool for identifying focalized hidden information in data where the coordinate axes involved have physical interpretations.

In our BoW approach to object recognition, each coordinate axis in the visual histogram corresponds to a particular visual word in the vocabulary tree. We contend that the visual words that explain maximum variability in data corresponding to each object category can be regarded as informative features for object recognition. In order to use Sparse PCA to identify these visual words, an empirical covariance matrix must be computed for each object category in the database.

Let us consider m available training images of an object category. Using the constructed vocabulary tree learned from all the categories, the SURF descriptors in each image are converted into a visual histogram $\boldsymbol{y} \in \mathbb{R}^n$. The m vectors $\{\boldsymbol{y}_j\}$ are then normalized to have unit length and centered, and grouped into a *data matrix*: $A = [\tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2, \cdots, \tilde{\boldsymbol{y}}_m] \in \mathbb{R}^{n \times m}$. The empirical covariance matrix is then computed from this data matrix as $\Sigma_A = \frac{1}{m} A A^T$.

Sparse PCA that computes the first sparse eigenvector of Σ_A optimizes the following objective [117]:

$$\boldsymbol{x}_s = \arg\max \boldsymbol{x}^T \Sigma_A \boldsymbol{x} \quad \text{subj. to} \quad \|\boldsymbol{x}\|_2 = 1, \|\boldsymbol{x}\|_1 \leq k. \tag{3.3}$$

We denote the indices of the non-zero coefficients in $\boldsymbol{x}_s$ by $\mathcal{I}$ (*i.e.*, the nonzero support of $\boldsymbol{x}_s$). These indices correspond to the visual words that explain maximum variability in A, and are subsequently used in the object recognition process (explained in section 3.6).

In practice, it is common that the leading first sparse PV may not be sufficient for obtaining a variable support, and it is desirable to further estimate a few subsequent sparse PVs as well. In optimization, it is a common practice to estimate succeeding eigenvectors by sequentially deflating the covariance matrix with the preceding ones. Several techniques have been explored for reliably deflating a covariance matrix for Sparse PCA [64]. We adopt a simple technique called Hotelling's deflation that eliminates the influence of the first sparse PV to obtain a deflated covariance matrix

Σ'_A as follows:

$$\Sigma'_A = \Sigma_A - (\boldsymbol{x}_s{}^T \Sigma_A \boldsymbol{x}_s)\boldsymbol{x}_s \boldsymbol{x}_s{}^T. \tag{3.4}$$

Then, the second sparse eigenvector $\boldsymbol{x}'_s$ of Σ_A becomes the leading sparse eigenvector of Σ'_A, and can be estimated again by Sparse PCA (3.3). In our experiment, we observe that the first two sparse PVs are sufficient for selecting informative features that lie on the foreground objects in the BMW database (as shown in Figure 3.1 and 3.4). Finally, If we denote the indices of the non-zeros in the second PV $\boldsymbol{x}'_s$ as $\mathcal{I}'$, then the union $\mathcal{I} \cup \mathcal{I}'$ provides the support corresponding to the informative features of a particular category.

We have also compared the variable selection performance of Sparse PCA with thresholded PCA. To obtain a thresholded PCA support set, we perform classical PCA on the same covariance matrix Σ_A and pick the top k indices of the corresponding first and second PVs with highest absolute value as the informative features. Here, k is chosen as the cardinality of the corresponding Sparse PVs for the same category. The third row of Figures 3.1 and 3.4 show the informative features selected by thresholded PCA (pink). It is clear in these figures that majority of the informative features do not lie on the foreground objects.

3.4 Speeding up Sparse PCA using ALM

Sparse PCA has been an active research topic for over a decade. Notable approaches include SCoTLASS [47], SLRA [115], and SPCA [117], all of which aim at finding modified PVs with sparse loadings. However, one drawback of all the above algorithms is that the formulation requires solving nonconvex objective functions. Recently, d'Aspermont *et. al.* [23] derived an ℓ_1-norm based semidefinite relaxation for Sparse PCA called DSPCA, and it is currently the most widely known convex formulation of the problem. This algorithm, however, has a slow convergence rate which is a major bottleneck when analyzing high dimensional data. Augmented Lagrange multiplier (ALM) based algorithms have recently gained a lot of popularity due to their rapid convergence and speed in ℓ_1-minimization [3] and Robust PCA [113] problems. These have motivated us to develop a new algorithm for solving the semidefinite relaxation form of Sparse PCA using ALM.

We begin by considering an empirical covariance matrix $\Sigma \in \mathbb{S}^n$, with n representing the dimensionality of the data. Sparse PCA can be formulated as:

$$\max_{\|\boldsymbol{x}\|_2 \leq 1} \boldsymbol{x}^T \Sigma \boldsymbol{x} - \rho \|\boldsymbol{x}\|_0, \tag{3.5}$$

where $\rho > 0$ is a scalar parameter controlling the sparsity in $\boldsymbol{x}$. By following the ℓ_1-norm relaxation and lifting procedure for semidefinite relaxation presented in [23], and dropping a nonconvex rank constraint, we can rewrite (3.5) as:

$$\max_X \mathbf{Tr}(\Sigma X) - \rho \|X\|_1 \ : \ \mathbf{Tr}(X) = 1, X \succeq 0, \tag{3.6}$$

[1] $\|X\|_1$ represents the entrywise norm: $\mathbf{1}^T |X| \mathbf{1}$.

where $X = \boldsymbol{x}\boldsymbol{x}^T$ is a matrix variable. Duality allows us to rewrite this problem as a SDP:

$$\min_U \ \lambda_{max}(\Sigma + U) \ : \ -\rho \leq U_{ij} \leq \rho. \tag{3.7}$$

As presented in [23], assuming Σ is fixed and given, the maximum eigenvalue function $\lambda_{max}(\cdot)$ can be approximated by a smooth, uniform objective (*i.e.*, with Lipschitz continuous gradient):

$$f_\mu(U) = \mu \log(\mathbf{Tr} \exp((\Sigma + U)/\mu)) - \mu \log(n), \tag{3.8}$$
$$\nabla f_\mu(U) = \exp((\Sigma + U)/\mu)/\mathbf{Tr}(\exp((\Sigma + U)/\mu)), \tag{3.9}$$

where $\mu = \epsilon/2\log(n)$ produces an ϵ-approximate solution. With this approximation, (3.7) can be rewritten as,

$$\min_U \ f_\mu(U) \ : \ -\rho \leq U_{ij} \leq \rho. \tag{3.10}$$

The basic idea of ALM methods is to eliminate the constraints and add to the cost function a penalty term that prescribes a high cost to infeasible points [8]. This augmented cost function is called the *augmented Lagrangian function*. In our case, the box constrained convex problem of (3.10) can be written in an unconstrained form as:

$$\min_U \{f_\mu(U) + \sum_{1 \leq i,j \leq n} P(U_{ij}, Y_{ij}, c)\}, \tag{3.11}$$

where $Y_{ij}, 1 \leq i, j \leq n$ represents the Lagrange variable, c determines the severity of the penalty, and

$$P(u, y, c) = \begin{cases} \textit{i.e.} y(u - \rho) + \frac{c}{2}(u - \rho)^2 & \text{if } \rho - \frac{y}{c} \leq u, \\ y(u + \rho) + \frac{c}{2}(u + \rho)^2 & \text{if } -\rho - \frac{y}{c} \geq u, \\ \frac{y^2}{2c} & \text{otherwise. } \textit{i.e.} \end{cases} \tag{3.12}$$

We denote as $F(U, Y)$ the cost function of (3.11), which is our smooth and convex augmented Lagrangian function with Lipschitz continuous gradient $\nabla_U F(U, Y)$.

The algorithm for Sparse PCA using ALM (SPCA-ALM) is presented in Algorithm 1. Note that in each iteration of the outer loop of the algorithm, we need to solve the unconstrained minimization problem in (3.11), which has no closed-form solution. Thus, we employ Nesterov's first order gradient technique [68]. Once this augmented Lagrangian function is minimized, the Lagrange multiplier Y will be updated using the rule:

$$Y_{ij}^{k+1} = \begin{cases} \textit{i.e.} Y_{ij}^k + c^k(U_{ij}^k - \rho) & \text{if } Y_{ij}^k + c^k(U_{ij}^k - \rho) > 0, \\ Y_{ij}^k + c^k(U_{ij}^k + \rho) & \text{if } Y_{ij}^k + c^k(U_{ij}^k + \rho) < 0, \\ 0 & \text{otherwise. } \textit{i.e.} \end{cases} \tag{3.13}$$

After the algorithm converges, the primal variable is given by the gradient in (3.9), *i.e.*, $X^k = \nabla f_\mu(U^k)$. Then the sparse principal component is recovered as the leading eigenvector of X^k.

Algorithm 1 SPCA-ALM

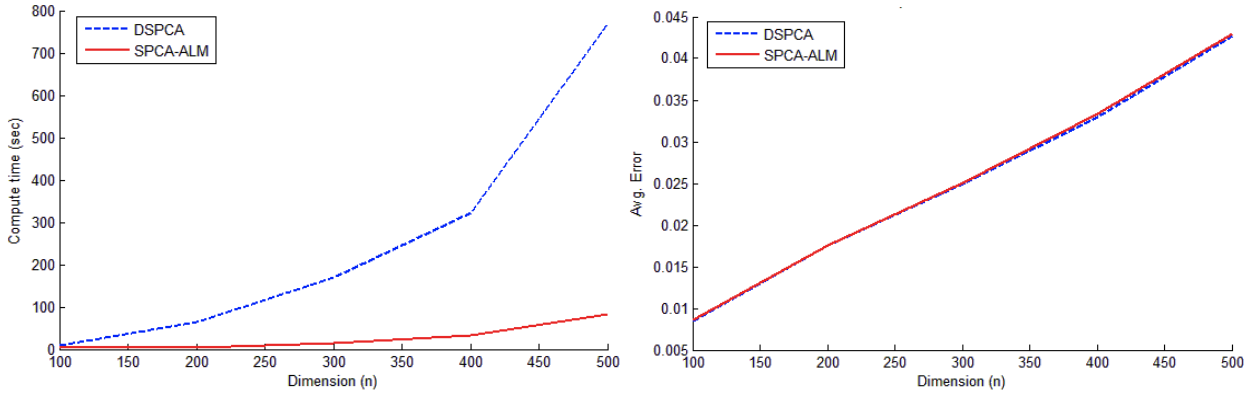
Input: Covariance Σ and $\rho > 0$.

$U^1 \leftarrow \mathbf{0}$, $Y^1 \leftarrow \mathbf{0}$, $X^1 \leftarrow \mathbf{0}$, $c^1 \leftarrow 1$.
while not converged (k=1,2,3,...) **do**
  $t^1 \leftarrow 1$, $V^1 \leftarrow U^k$, $W^0 \leftarrow U^k$, $Z \leftarrow rand(n, n)$.
  $\alpha^0 \leftarrow \frac{\|V^1 - Z\|_2}{\|\nabla F(V^1, Y^k) - \nabla F(Z, Y^k)\|_2}$.
  while not converged (l=1,2,3,...) **do**
    Find smallest $i \geq 0$ for which
    $F(V^l, Y^k) - F(V^l - \frac{\alpha^{l-1}}{2^i}\nabla F(V^l, Y^k), Y^k) \geq \frac{\alpha^{l-1}}{2^{i+1}}\|\nabla F(V^l, Y^k)\|_2$.
    $\alpha^l \leftarrow 2^{-i}\alpha^{l-1}$, $W^l \leftarrow V^l - \alpha^l \nabla F(V^l, Y^k)$.
    $t^{l+1} \leftarrow (1 + \sqrt{4t^{l^2} + 1})/2$.
    $V^{l+1} \leftarrow W^l + \frac{t^l - 1}{t^{l+1}}(W^l - W^{l-1})$.
  end while
  $U^{k+1} \leftarrow W^l$
  Update Y^{k+1} using the update rule (3.13).
  $X^{k+1} \leftarrow \nabla f_\mu(U^{k+1})$.
  $c^{k+1} \leftarrow 2^k$.
end while

Output: Sparse principal vector, $x_s \leftarrow$ leading eigenvector of X^k.

3.4.1 Performance

We have evaluated our SPCA-ALM algorithm by comparing its performance against the DSPCA solver [23]. Both algorithms have been implemented in MATLAB and benchmarked on a 2.6 GHz Intel processor with 4 GB memory. We generate synthetic data of varying dimensionality as follows. First, in the n-dimensional vector space, 10% of its indices are selected as nonzero support. Next, the values of the nonzero coefficients are drawn from an independent and identically distributed Gaussian $\boldsymbol{x}_0(i) \sim N(0, 200)$. Finally, random noise $\epsilon \sim N(0, 1)$ is added to x_0 to form a noisy version of the empirical covariance matrix, $\Sigma = (\boldsymbol{x}_0 + \epsilon\mathbf{1})(\boldsymbol{x}_0 + \epsilon\mathbf{1})^T$. This covariance matrix, along with an optimal choice of the parameter ρ to encourage sparsity, is provided to both the SPCA-ALM and DSPCA algorithms. The process repeats 10 times for each problem dimension n, while n varies from 100 to 500 and the mean speed and precision are computed for each n. Figure 3.2a compares the speed of the two algorithms, while Figure 3.2b compares the estimation error of the first estimated sparse principal vector. The simulation shows SPCA-ALM converges much faster than DSPCA (for example, at $n = 500$, SPCA-ALM is about 10 times faster), while maintaining approximately the same reconstruction accuracy.



(a) Speed vs Data Dimension

(b) Estimation Error vs Data Dimension

Figure 3.2: A comparison of SPCA-ALM and DSPCA using simulated data.

3.5 Variable Elimination via SAFE

In this section, we further examine a dimensionality reduction technique as a preprocessing step to speed up Sparse PCA. Particularly in object recognition, the covariance matrix Σ often can be of high dimension (*e.g.*, 1000 and higher). Directly calling SPCA-ALM may still be very time consuming. To mitigate this problem, we invoke a feature elimination method presented in [108, 25], called SAFE. The method allows to quickly eliminate variables in problems involving a convex loss function and a ℓ_1-norm penalty, thereby leading to substantial reduction in the number of variables prior to running optimization. The following Theorem [108, 25] states the SAFE method applied to Sparse PCA. An illustration of this process is shown in Figure 3.3.

Theorem 1 (SAFE Variable Elimination for Sparse PCA)**.** *Given a covariance matrix Σ, denote σ_k as its kth diagonal entry. For the Sparse PCA problem (3.5), if $\rho > \sigma_k$, then the kth element of the solution $\boldsymbol{x}_s$ will never be in the support. Hence, the kth row and column of Σ can be removed from the optimization.*

Therefore, for a predefined choice of ρ, we first obtain a reduced covariance matrix by eliminating all the rows and columns corresponding to those variables with sample variance less than ρ. The number of variables thus eliminated is a conservative lower bound on the total number of zero-weight variables in the final solution of Sparse PCA. In our experiments, we typically can eliminate about 90% of the variables using SAFE without sacrificing the accuracy of preserving important informative features.

3.6 Experiments

In order to test the effectiveness of suppressing uninformative features for the task of object recognition, we have evaluated the performance of our method on the Berkeley Multiview Wireless

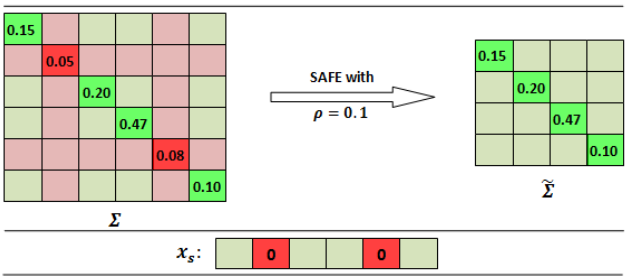


Figure 3.3: SAFE feature elimination process. **Top:** The red rows and columns of a sample covariance matrix Σ are eliminated to form new covariance matrix $\tilde{\Sigma}$, as the corresponding variances are less than chosen $\rho = 0.1$. **Bottom:** The loadings of the corresponding indices are subsequently zeroed out in x_s.

(BMW) database [67]. The database consists of multiple-view images of 20 landmark buildings on the campus of University of California, Berkeley. For each building, wide baseline images have been captured from 16 different vantage points. Further, at each vantage point, 5 narrow baseline images have been captured, thereby summing to 80 images per category. All images are 640×480 RGB color images. It is important to note that the image quality in this database is considerably lower than many existing high-resolution databases, which is intended to reproduce realistic imaging conditions for mobile camera and surveillance applications. Further, it is noticeable that some images are slightly out of focus and in some cases, even corrupted by dust residual on the camera lenses.

We divide the database into a training set and a testing set. The vantage points of each object are named numerically from 0 to 15. All these 16 images of each category captured from camera #2 are designated as the training set, and the remaining images are assigned to the testing set. Thus, there are 16 training images and 64 testing images for each category. We extract SURF keypoints in each of the images and construct a vocabulary tree with 1000 leaf nodes using the keypoints descriptors from all the training images.

3.6.1 Results

We first evaluate the recognition accuracy of the classifier (3.2) without suppressing any features from the training and testing sets to obtain a baseline performance. The results of this experiment are presented in Table 3.1. For the 20 object categories tested, the average baseline recognition rate is around 80%.

Next, for each object category i, we obtain its corresponding visual word support set $\mathcal{I}_i$ by determining the indices of the non-zero variables in the first and second sparse PVs. These are estimated by running Sparse PCA on the covariance matrix corresponding to the training histogram vectors in ith category. We then form the total support set $\mathcal{I}_{\text{SPCA}}$ for the entire database by taking the union of all the individual visual support sets for all the 20 object categories, *i.e.*,

$$\mathcal{I}_{\text{SPCA}} = \mathcal{I}_1 \cup \mathcal{I}_2 \cup \cdots \cup \mathcal{I}_{20}.$$

Figure 3.4: **Top:** Images of 6 objects in the BMW database with superimposed SURF features; **Middle-top:** Informative features detected by the SfM approach; **Middle-bottom:** Informative features detected by thresholded PCA; **Bottom:** Informative features detected by Sparse PCA (given by first two leading sparse PVs).

In our experiments, we have set the sparsity controlling parameter ρ to 0.002 for all the categories. With this choice of ρ, at roughly 20 variables per category, our total support set $\mathcal{I}_{\text{SPCA}}$ identifies 405 informative features, thereby rejecting a fraction of $\frac{3}{5}$ of the visual words from the 1000-D vocabulary. With this subset of visual words, we further evaluate the recognition accuracy of (3.2) again. The results of this are also presented in Table 3.1. As one can see, for most of the categories, there is a significant improvement in the recognition accuracy, leading to the average recognition rate at 85%, 5% higher than the baseline.

For some of the object categories, the SfM method (section 3.2.1) does seem to work reasonably well, and with these categories, we have formed a SfM index set, $\mathcal{I}_{\text{SFM}}$. We have tested the recognition accuracy of these visual words on the database as well and we have obtained an average rate of 78%. It can be seen in the fifth column of table 3.1 that the performance of the SfM method is inferior to that of the SPCA method for almost all object categories. Some visual comparisons between the results from Sparse PCA and SfM are presented in Figure 3.4.

3.7 Conclusion and Discussion

We have presented a novel and effective solution to select informative features for object recognition by Sparse PCA. For applications that involve low-quality mobile cameras or camera sensor networks, existing SfM solutions to detect and suppress uninformative features tend to fail. We have shown that Sparse PCA can successfully identify important visual features that explain max-

imum variability in the visual histogram vectors. For our database, these features correspond to those visual words that most often represent the appearance of foreground objects. To further speed up the execution of Sparse PCA, we have developed an improved numerical solution, namely, ALM. The new algorithm has proved significantly faster than the other convex semidefinite programming solutions. Using a public multiple-view image database, our experiment shows the estimated informative features improve the overall recognition rate by 5% compared to the baseline solution, and by 7% compared to the SfM solution.

For future work, we believe the two existing approaches, namely, Sparse PCA and SfM, are complementary under more general object recognition settings, which may lead to further improvement of the performance. We would like to focus on further combining our batch numerical technique with a geometric RANSAC scheme to robustly detect informative features in both low-quality and high-quality image databases.

Table 3.1: Recognition rates for all object classes. The best rates are marked in bold face. The number of informative features chosen per category are presented in the fourth and last columns for Sparse PCA and SfM respectively. The categories for which SfM failed have 0 informative features in the last column.

Cat.	Baseline Rate(%)	Sparse PCA Rate(%)	Sparse PCA # Feat	SfM Rate(%)	SfM # Feat
Bo	**98.61**	94.44	35	83.33	0
Cal	90.27	**91.66**	23	90.27	35
Cam	56.94	**66.66**	15	58.33	0
EAL	70.83	**81.94**	12	65.27	30
Ev	77.77	**91.66**	56	81.94	0
FH	**95.83**	88.88	23	87.50	0
G	79.16	**93.05**	34	86.11	0
Haas	77.77	**91.66**	30	72.22	0
HG	56.94	**73.61**	45	63.88	11
HM	51.38	**65.27**	9	61	0
Hg	**83.33**	76.38	76	69.44	13
HMC	81.94	**83.33**	28	70.83	0
LC	62.50	**72.22**	43	52.77	0
MaL	**98.61**	93.05	20	90.27	37
MuL	69.44	**80.55**	36	75.00	0
PL	58.33	79.16	53	**80.55**	66
SG	**100.00**	90.27	17	84.72	0
Sp	98.61	93.05	45	**100.00**	56
VLSB	**97.22**	83.33	24	86.11	0
Wu	98.61	**100**	46	95.83	0
Avg.	80.02	**84.51**	33	77.77	36

Chapter 4

Joint Detection and Categorization of Human Actions

4.1 Introduction

Traditional Closed Circuit TV (CCTV) camera based surveillance systems typically consist of several wired cameras distributed within a building and the surrounding site, transmitting video streams to a control room as shown in Fig. 4.1. The security personnel employed are expected to monitor activity on all the video feeds, due to which several events can go unnoticed. Further, for applications such as indexing and retrieval of surveillance video, manual methods can be extremely time consuming, monotonous and stressful.

Figure 4.1: Typical CCTV control room with video feeds from several cameras

In the computer vision and robotics research communities, on the other hand, significant progress has been made in the areas of action recognition [9]-[84]. Most of this work has focussed on automatically recognizing human actions in publicly available datasets composed of pre-segmented video clips [9, 84]. The methods developed have primarily addressed variability in scale of the subjects, background clutter suppression and handling occlusions in the video clips [96]. These methods, however, are not directly applicable to surveillance systems as the temporal segmentation of continuous video is a challenging problem. Some recent works have focussed on partitioning the temporal segmentation and recognition process for long video sequences [57, 83]. However, these methods perform poorly, as low-level temporal cues are generally not discriminative enough for precisely partitioning the video. Some algorithms have also been developed for detecting and recognizing actions in generic video sequences [69, 92, 56]. These methods typically require a lot of processing at the image level, therefore making them hard to implement on a wireless smart camera.

In this chapter, we present a novel system for simultaneous detection and recognition of human actions in wireless smart camera networks. This system is an extension of the distributed object recognition system presented in Chapter 2. Our system is capable of handling video sequences captured by a single camera or multiple cameras with overlapping views. It is partitioned into distributed feature extraction (performed on the wireless smart cameras) and centralized spatiotemporal multi-view activity detection and recognition (performed at a base station). Each wireless camera in our system is capable of extracting, encoding and transmitting a descriptor vector corresponding to foreground objects of interest in every frame where motion is detected. At the base station, descriptor vectors from a single or multiple camera sources are fused within a graphical model framework for localizing and recognizing actions of interest. Our graphical model framework is based on the famous Deformable Part Models (DPMs) for object detection in static images proposed by Felzenszwalb *et al*. [33]. We have extended the DPM framework to the spatiotemporal setting for both single and multiple view video streams. At its core, our algorithm replaces part appearance templates of the DPM by class-specific keyframes, and enforces spatiotemporal constraints between pairs of keyframes in the single-view setting. In the multiple-view setting, homography constraints [63] induced by the ground plane are used to enforce spatial connectivity between object regions in images from pairs of cameras.

The exposition of this chapter is as follows. In section 4.2 we provide a brief literature review of activity recognition, while focussing on recent work that address similar problems as ours. We present our overall system pipeline in section 4.3 and discuss those basic elements of our pipeline that are drawn from previous work. The primary contribution of our chapter is the centralized, multi-view spatiotemporal action detection and recognition algorithm and is presented in section 4.4. We validate the performance of our algorithm by performing experiments on standard and novel datasets, as presented in section 4.5. Section 4.6 provides a conclusion and an outline for future work.

4.2 Literature Review

Spatiotemporal bag-of-word representations are amongst the most popular approaches for action recognition because of their ease of use, and high discriminating capabilities [61, 84]. They have successfully been employed in both single view [57, 70, 101] and multi-view settings [107]. Although they work very well on temporally segmented video clips, they cannot be extended to action detection directly, as they ignore spatial and temporal relationships between discriminative templates. Further, detecting and describing spatiotemporal interest points would require significant processing, which can be a challenge for a low-power smart camera.

Other spatiotemporal template and filter based methods have also gained significant traction for action detection and recognition. Gorelick *et al.* [9] extract foreground silhouettes of moving people and use them to construct volumetric features for action recognition. Rodriguez *et al.* [79] use MACH filter responses to detect actions of interest. Ali & Shah [6] extract kinematic features from images to recognize actions. [96] provides an excellent survey of similar state-of-the-art methods. All these methods, however, require features extracted from every frame in a temporal volume. Thus, they would not work well within our framework, where the frequency of sampling images and transmitting extracted features needs to be variable in order to accommodate varying bandwidth constraints.

In the image based human pose and object detection literature, DPMs have gained a lot of popularity [33, 109]. Such human pose detectors have been fused with traditional image segmentation techniques to extract foreground pixels corresponding to people in static images [65]. Niebles *et al.* adapted the DPM framework to temporal action detection [69]. Tian *et al.* [92] and Lan *et al.* [56] have extended the framework to the spatiotemporal setting. While these methods are similar in spirit to our algorithm, their focus is on generic videos where no assumptions can be made regarding the background. Thus, these methods are very computationally intensive and cannot be easily adapted to wireless surveillance applications. Further, it becomes exponentially complex to extend their inference algorithms to multi-view scenarios, even after incorporating epipolar constraints.

Generative methods for activity recognition have been extensively addressed by the computer vision and control community. Sminchisescu *et al.* [89] have proposed conditional models for human action recognition. Fox *et al.* [105] and Tao *et al.* [91] have used Hidden Markov Models with Dirichlet and sparsity priors respectively for action and gesture recognition. Niebles *et al.* [70] have used Probabilistic Latent Semantic analysis for learning human actions in an unsupervised setting. Wang *et al.* [102] have used HMMs to recognize actions performed by gymnasts in multi-view settings. 3D exemplar based HMMs are used by Weinland *et al.* [103] to recognize actions in arbitrary views of camera networks. All these generative methods tend to perform poorly in the presence of actions that are not pre-defined during training. Further, in real surveillance settings, transition probabilities are very hard to estimate as different people being tracked might have different goals and destinations.

Some recent works on joint segmentation and recognition of human actions have addressed a problem related to ours. Shi *et al.* [85] introduce a Semi Markov model framework to capture the temporal structure of actions in video sequences. They present a structured learning framework to learn the parameters of their graphical model, and a Viterbi-style inference algorithm that works

real-time on long video sequences. Hoai *et al.* [40] employ a similar approach with a multi-class SVM for learning model parameters and a slightly different cost function for inference. While these methods can be extended to our wireless surveillance camera setting, they still require transmission of every frame captured by the camera sensor, which can be challenging in resource constrained settings. Further, since their framework is purely temporal, multi-view information across pairs of cameras cannot be easily utilized.

4.3 System Pipeline

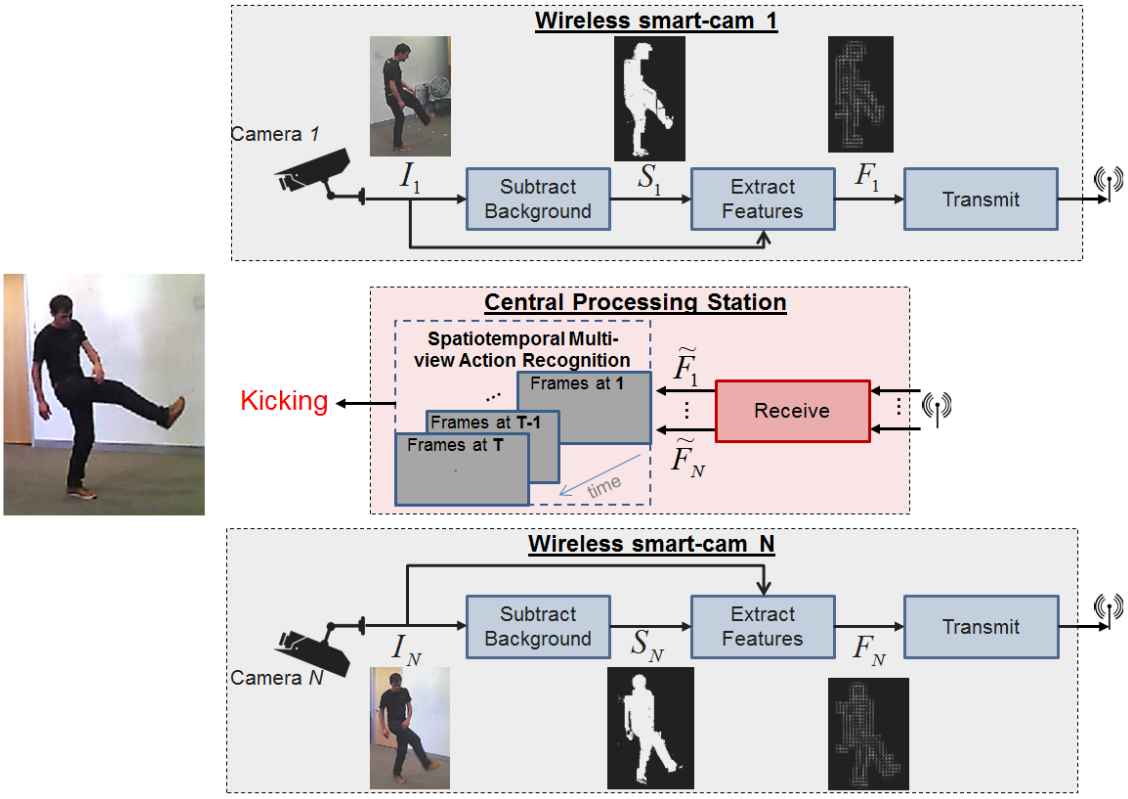


Figure 4.2: System pipeline. See text for details.

Our system consists of multiple smart cameras communicating wirelessly with a central processing station as shown in Fig. 4.2. In our current framework, we assume that all the cameras connected to the base station are viewing the same scene from different vantage points, and that images from all of them share some amount of overlap. We also assume that the cameras are time synchronized, and that minimal extrinsic calibration is available between pairs of cameras. The details of our calibration and the spatiotemporal multi-view recognition algorithm of the central processing station are presented in the section that follows.

Each wireless smart camera is capable of separating foreground objects, extracting gradient features for each object, and transmitting these features to the central processing station. We use an off-the-shelf background subtraction algorithm [116] to extract foreground object silhouettes in each camera. The size of the bounding box around the object can be used to determine the scale of the object. Nonetheless, it is impossible to uniquely disambiguate the size of the object and its distance from the camera as this information is lost during perspective projection. For our activity recognition application, however, we argue that the scale encodes sufficient information, as the distance covered by a smaller person translating closer to a camera can be comparable to that covered by a larger person further away.

Some activity recognition papers use features computed on silhouettes as inputs to their algorithms [9, 85, 40]. However, due to self occlusion, discriminative details within the object boundary can be lost when using silhouettes. For instance, this can be seen in the silhouette extracted by the first camera in Fig. 4.2, where the arm of the person is fully encapsulated by the boundary around his silhouette. In order to utilize maximum information available in each image, we extract HOG descriptors [22] within the bounding box around the foreground object. Specifically, we use the silhouette to extract the foreground pixels within the bounding box, and apply a grid to the foreground region. The number of rows and columns of the grid are kept constant for all foreground regions. In our experiments we have used 5×5 grids for each foreground object. HOG descriptors are extracted within each grid and vectorized to represent the appearance of the foreground object. These appearance descriptors along with the bounding box coordinates are subsequently transmitted wirelessly to the central processing station.

System Analysis: The processing performed on board each wireless camera is largely stabilized, making it amenable to deployment with minimal requirements for firmware updates. Even in situations where the number of action classes or the entire action recognition framework changes, the basic operations performed on the smart camera can remain unaltered. The primary purpose of feature extraction on-board the camera is to minimize the data transmitted. In the current framework, only 800 bytes ($5 \times 5 \times 32$) of data is transmitted for every object detected. Further, we can leverage the sparsity of the feature descriptors and utilize a compression scheme similar to that presented in [40]. In comparison, H.264 video compression provides an average bit rate of 64K bytes per image for 640×480 color images [104] (roughly 2 orders of magnitude higher) with more complex processing performed on the imaging platform. Although this analysis assumes that only one object is detected in any frame, this is still a conservative estimate of transmission savings, as there are going to be situations where no people are present or moving in front of the cameras. This leads us to believe that our system is an attractive wireless alternative for automated surveillance applications.

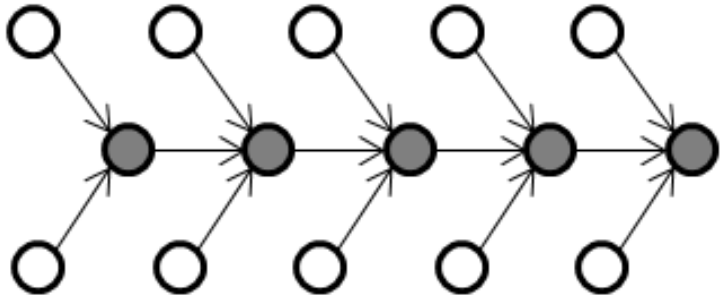


Figure 4.3: Multi-view graphical model that represents any particular action. Filled nodes represent keyframes in reference camera, and empty nodes represent keyframes in other two cameras.

4.4 Spatiotemporal Multi-View Action Recognition

4.4.1 Deformable Keyframe Model (DKM)

Single-view Model: Our keyframe based action detection framework is closely related to the DPM model commonly used for object detection [33]. We represent a video sequence as D and any particular action as an N node directed graph, $G = (V, E)$. The nodes in the graph, V, correspond to keyframes. Any given node $i \in \{1...N\}$ has an anchor position $p_i = (x_i, y_i, t_i)$, where (x_i, y_i) represent the pixel location of the center of the bounding box around an object in the image, and t_i represents the frame number in the video sequence. Edges in the graph, E, specify which pairs of keyframes are constrained to have relations. The framework is very general and edges in the graph need not be successive. For instance, jump edges can be used to connect nodes corresponding to repetitive keyframes in cyclical actions.

The score, $\boldsymbol{S}$, associated with a particular action model and keyframe-labeling can be written as [33]:

$$\boldsymbol{S}(p|D, \mathbf{w}) = \sum_{i \in V} \langle w_i, \phi^{app}(D, p_i) \rangle + \sum_{i,j \in E} \left\langle w_{ij}, \phi^{def}(p_i, p_j) \right\rangle \tag{4.1}$$

where, $\phi^{app}(D, p_i)$ is the HOG appearance descriptor of the object detected at frame t_i (see section 4.3 for details), and $\phi^{def}(p_i, p_j)$ models the deformation between pairs of frames. In the single-view setting, the deformation is given by $\phi^{def}(p_i, p_j) = [dx, dx^2, dy, dy^2, dt, dt^2]$, where $dx = x_i - x_j$, $dy = y_i - y_j$ and $dt = t_i - t_j$. For the right match, the keyframe appearance template, w_i, will have a maximum inner product response with the appearance descriptor at location p_i in the video D. The deformation weight w_{ij} models the Mahalanobis distance between the pairs of keyframes in the model, and its parameters need to be learned during training. We address the learning of appearance and deformation weights in section 4.4.3.

Multi-view Model: We extend our single-view keyframe model framework to incorporate multiple cameras capturing the same scene. In this case, we choose one reference camera, and all other cameras are connected to it, thereby yielding a directed graphical model as shown in Fig. 4.3. In this chapter, we do not model the spatiotemporal relationship between nodes corresponding to each

Figure 4.4: Deformation constraints between reference view in the middle and two other cameras viewing the scene. Deformation cost modeled as spring connecting center of line between bottom corners of each bounding box, as they lie on the ground plane. All three images are captured at same time instant from three vantage points.

non-reference camera. This would, however, be a straightforward extension as the introduction of spatiotemporal edges in non-reference cameras would not introduce any cycles in our directed graph.

The score function for the multi-view setting remains the same as that in the single-view model in eqn. 4.1. The deformation function between frames captured at the same time instance from two views, however, needs to account for the epipolar constraints between the views. In most surveillance settings, it is common to have significant overlapping views of the ground plane on which people move about. We use the homography induced by this ground plane to enforce pairwise constraints between views.

Specifically, we compute the ground plane homography, H_l^r, between any camera l in the network that shares scene overlap with the reference camera r. Since the homography is a linear transform that maps pixels in one view of a plane to another, it can be used to determine the distance between object detections across views. Further, since the people in the cameras' fields-of-view are in contact with the ground plane at most times, the centre of the line connecting the bottom corners of the bounding box detection around them can be used as a proxy for their 3D location in the scene. Although this assumption can be easily violated when people are closer to the camera, in surveillance applications, that is unlikely as cameras are intentionally positioned far from reach.

Given a pixel $p^l = (x^l, y^l, 1)^T$ on the ground plane in the l^{th} camera view, its position in the reference camera can be estimated as $\tilde{p^r} = H_l^r p^l$. The deformation function for the two views can then given by $\phi^{def}(p_i^l, p_i^r) = [dx, dx^2, dy, dy^2]$, where, $[dx, dy] = (p^r - H_l^r p^l)^T$. Fig. 4.4 shows an example with deformation constraints between a reference camera and two other cameras on either side of it.

4.4.2 Keyframe Selection

Analogous to parts in DPMs for object detection, our deformable keyframe models use appearance templates corresponding to keyframes as node potentials. Thus, it is important for the same set of keyframes to be present in all samples of a given action, at least while learning the model parameters. We adopt the definition proposed by Bourdev & Malik [10] to define keyframes in

our setting: Given a set of M training video samples $\{D_1, ..., D_M\}$ of any action, the goal of keyframe selection is to find a subset of N representative frames in each sample such that, similarly selected representative frames of actions are tightly clustered in 3D body configuration space. This process of supervised clustering of keyframes can easily be done using motion capture, where different subjects perform actions while simultaneously being recorded by a motion capture system to capture their 3D pose and a camera network to capture their appearance in multiple views. Using this method, we can automatically obtain ground-truth keyframe labelings $\{p_1, ..., p_M\}$, for all the video samples. In our experiments, however, we have manually annotated the keyframes as we were unable to find any publicly available complex action datasets captured using motion capture and traditional cameras.

4.4.3 Learning

We employ a structured learning [94] approach to train the parameters of our model for each action, $c \in \{1...C\}$, where C is the total number of actions in our database. Given a set of M positive training examples $\{D_q\}$ $(q = 1, 2, ...M)$ for any action c, we are interested in learning the appearance (w_i^c's) and deformation parameters (w_{ij}^c's) given in eqn. 4.1 that would produce the correct labeling $\{p_q\}$. Since our scoring function (4.1) is linear in these parameters, it can be rewritten as

$$\boldsymbol{S}(p_q|D_q, \mathbf{w^c}) = \langle \mathbf{w^c}, \Phi(D_q, p_q) \rangle \,, \tag{4.2}$$

where, $\mathbf{w^c}$ is a vector that includes all the appearance and deformation parameters that need to be learned, and $\Phi(D_q, p_q)$ is the corresponding appearance and deformation energy due to a certain labeling p_q.

In our setting, we are also interested in discerning different actions from each other, so we need to learn models that can jointly detect and discriminate between different actions. We adopt a one-vs-all learning policy for each action, and learn the model parameters that can jointly detect and recognize any particular action given hard negative examples of other actions in the database.

We adopt the structural SVM framework of [94] and write our learning objective as,

$$\begin{aligned} &\underset{\mathbf{w^c}, \{\xi_q\}, \{\eta_{q,q'}\} \geq 0}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{w^c}\|^2 + \lambda_1 \sum_q \xi_q + \lambda_2 \sum_{q,q'} \eta_{q,q'} \\ \text{s.t.} \quad &\forall q, \langle \mathbf{w^c}, \Phi(D_q, p_q) - \Phi(D_q, \tilde{p}) \rangle \geq \Delta(p_q, \tilde{p}) - \xi_q \\ &\forall q, q', \langle \mathbf{w^c}, \Phi(D_q, p_q) - \Phi(D_{q'}, p_{q'}) \rangle \geq \Delta(p_q, p_{q'}) - \eta_{q,q'}, \end{aligned} \tag{4.3}$$

where, λ_1, λ_2 are user defined scaling parameters to minimize slack values in the optimization.

The first constraint in eqn. 4.3 implies that for the same class, any keyframe labeling $\tilde{p}$, other than the ground-truth labeling p_q, for the q^{th} data sample, needs to be penalized according to the loss function $\Delta(p_q, \tilde{p})$. The non-negative slack term ξ_q provides an extra level of robustness to account for some violation of the constraint. The second constraint implies that given any ground truth labeling p_q for the q^{th} sample of a particular action, any ground truth labeling $p_{q'}$ of the $\{q'\}^{th}$ sample of any other action sequence in the database will produce a lower score after filtering through another violation accommodating hinge-loss $\eta_{q,q'}$.

The objective of the loss function $\Delta(p_q, \tilde{p})$ is to reflect how well a particular labeling hypothesis $\tilde{p}$, matches the true labeling p_q. We have adopted a simple binary loss function with $\Delta(p_q, \tilde{p}) = 1$ if $\tilde{p} = p_q$, and $\Delta(p_q, \tilde{p}) = 0$ otherwise. We employ the cutting-plane algorithm described in [94] to solve our quadratic program (4.3).

Model bias: The learning procedure, however, does not produce weights of the same magnitude for each action class. Thus, the modeling score for each action class has an associated bias b^c, that needs to be estimated and subtracted from the final score during inference. In order to determine the bias for each action class, we apply the learned model for that action class to the training data samples and take the median of these scores as the bias, i.e.,

$$b^c = \text{median}\{\boldsymbol{S}(p_1|D_1, \mathbf{w^c}), ..., \boldsymbol{S}(p_M|D_M, \mathbf{w^c})\}. \tag{4.4}$$

4.4.4 Inference

In our detection and recognition setting, given a query video sequence D, the inference problem is to find the best action c^*, and correspond labeling p^*, that maximizes the modeling score:

$$\{c^*, p^*\} = \underset{p, c \in \{1...C\}}{\text{argmax}} \quad (\boldsymbol{S}(p|D, \mathbf{w^c}) - b^c). \tag{4.5}$$

Since our directed graph is a chain in the single-view and a tree in the multi-view scenarios, inference can efficiently be done via dynamic programming [33].

4.5 Experiments

We evaluate our Deformable Keyframe Model (DKM) framework in three scenarios. In the first scenario, we test the discriminating capabilities of our model by performing whole-clip recognition. In the second scenario, we test the joint detection and recognition capabilities of our model in a controlled setting by synthesizing a complex action sequence by concatenating simple action video-clips. In the final scenario, we test our algorithm for joint detection and recognition of actions on a novel complex data set consisting of continuous actions performed by different subjects while being recorded by cameras placed at multiple vantage points.

4.5.1 Weizmann Simple Actions

The Weizmann dataset [9] is a popular dataset for validating action recognition algorithms, as it consists of short video clips captured under controlled conditions. It is composed of 10 action clips performed by 9 actors, all of whom remain un-occluded and at the same distance from the camera's focal plane. The background model of the scene is available, using which foreground silhouettes of the actors have been extracted for every frame of the video.

Keyframe selection: Automatic keyframe selection for the Weizmann dataset is challenging as there is no motion capture data available. Hence, we have manually selected keyframes for

each action. A set of 5 keyframes have been manually selected for each action performed by every individual.

We follow the same testing procedure proposed by [9] for the dataset. As presented in section 4.3, we extract HOG appearance descriptors for the foreground region in each frame, along with the coordinates of the bounding box. We use these features within our DKM framework and pick the action class that maximizes the modeling score (see eqn.4.5). Our DKM framework achieves **100%** recognition accuracy. This is comparable to the perfect recognition recognition reported by the authors of the dataset, and others who have also validated their methods after adopting the same testing procedure [9, 92, 28].

4.5.2 Weizmann Complex Actions

In order to validate the joint detection and recognition capability of our DKM, we synthesize complex actions by concatenating all the 10 actions performed by each of the subjects in the Weizmann dataset, thereby yielding 9 videos. The order in which the actions are composed is chosen at random for each subject. The frame level features are still extracted using the method outlined in section 4.3.

Our training methodology is similar to that employed by Hoai *et al.* [40]. We adopt a leave-one-out evaluation strategy: training on 8 sequences and testing on the left-out sequence. The models and associated bias for each action are learned using the procedure outlined in section 4.4.3.

Our evaluation metric is also inspired by theirs. Specifically, we evaluate each of our models on a query synthesized video. Multiple detections are found by each action specific DKM, and all the overlapping detections with the highest score per class are retained. The temporal union of these detections provides a class specific segmentation of the query video sequence. At this point, the overall frame-level accuracy against the ground truth labels is calculated as the ratio of number of agreements over the total number of frames. It is important to note that this segmentation based metric is designed for joint segmentation and recognition algorithms such as [40] and it serves as a harder baseline evaluation metric for our detection and recognition algorithm.

Fig. 4.5 shows the confusion matrix for the joint segmentation and recognition of the 10 actions using the 9 complex video sequences. The average accuracy of our method is **86.28%**. Hoai *et al.* [40] report an average accuracy of 87.7%, which is just slightly higher than our accuracy. However, their focus is on joint segmentation and recognition, and their algorithm yields a label for every frame in the query video. In our detection based framework, there is no guarantee that all frames will be assigned a class label, as evidenced by the white regions in our qualitative segmentation results shown in Fig. 4.6. This leads us to believe that our method will perform well even in the presence of previously unseen action classes, but we have not yet tested this hypothesis.

4.5.3 Bosch Multi-view Complex Actions (BMCA) Dataset

In the literature, there exist several public datasets for activity recognition, but continuous action datasets for action detection are limited. Further, to the best of our knowledge, there are no publicly available multi-view action detection datasets with subjects performing several actions

	walk	jack	jump	pjump	run	side	skip	walk	wave1	wave2
walk	0.87	0.13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
jack	0.00	0.95	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00
jump	0.00	0.00	0.90	0.00	0.00	0.00	0.10	0.00	0.00	0.00
pjump	0.00	0.00	0.00	0.90	0.00	0.00	0.00	0.10	0.00	0.00
run	0.00	0.00	0.00	0.00	0.90	0.00	0.00	0.00	0.10	0.00
side	0.00	0.05	0.06	0.00	0.00	0.89	0.00	0.00	0.00	0.00
skip	0.00	0.00	0.00	0.29	0.00	0.00	0.71	0.00	0.00	0.00
walk	0.00	0.00	0.00	0.21	0.07	0.00	0.00	0.72	0.00	0.00
wave1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.79	0.21
wave2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

Figure 4.5: DKM performance on the Weizmann complex dataset. Confusion matrix shows joint segmentation and recognition accuracy of 10 actions at frame level. Off-diagonal numbers show frame misclassification rates. Average accuracy of 86.28% achieved on dataset.

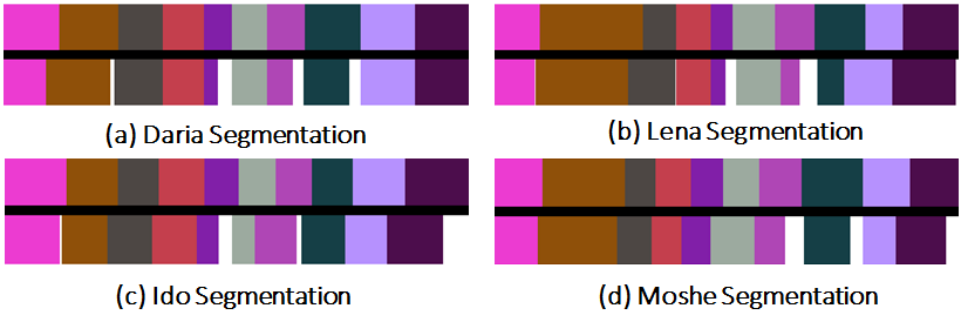


Figure 4.6: Qualitative segmentation of four complex videos. For each segmentation, top row shows true class labels and bottom row shows estimated labels. Note the existence of white regions in the estimated labels at frames where no reliable detections were found. As expected, majority of the error occurs at segment boundaries. Image best viewed in color.

continuously. To aid in peer evaluation of distributed activity detection and recognition, we have constructed a multi-view video dataset called the BMCA dataset which will be available online.

The BMCA dataset consists of 11 actions performed back-to-back by 20 subjects. Each subject performs three to four trials of each action while facing different directions, and at different locations within the capture area. The subjects are continuously recorded using 4 time synchronized cameras arranged in the configuration shown in Fig. 4.7. The cameras capture color video at a frame rate of 10 Hz, thereby yielding 4 long video clips of roughly 12-15 minutes each. The location within the capture area and the direction to face while performing an action are decided by the subjects themselves. However, they are all instructed to maintain angular orientations of roughly $\{0^o, 90^o, 180^o \text{ and } 270^o\}$ relative to the reference camera. In our setting, camera-2 is chosen as the reference view.

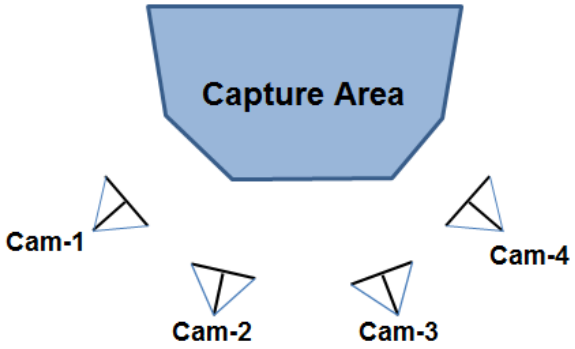


Figure 4.7: Configuration of cameras used to create BMCA dataset. Cameras capture color video at 10HZ, and are time synchronized.

Keyframe selection: The background subtraction scheme presented in section 4.3 has been used to obtain bounding boxes around people in the dataset. We have manually annotated the dataset by providing the start and end times of each action and its associated action class labels. The keyframes for each action have also been manually selected. Fig. 4.10 shows the keyframe annotations of 3 subjects performing 3 different actions.

Training: We have partitioned our dataset of 20 people into 5 training and 15 test sets. The 5 training sets include 11 actions performed at 4 angular orientations. Thus, we have learned 44 DKMs using the framework presented in section 4.4.3. We learn separate models for the single-view and multi-view experiments.

Testing: In order to validate our framework, we test our trained models on the 15 remaining test sets. As in the experiment for the Weizmann complex dataset, we employ the joint segmentation and recognition evaluation strategy. We only modify the segment labeling slightly so that all the detections corresponding to different orientations of the same action class are assigned the same label. We first evaluate the single-view DKM algorithm on the training videos captured by the reference camera. The results of our method is presented in the confusion matrix of Fig. 4.8. We obtain an average segmentation accuracy of **66.74%**. Although this accuracy is lower than that obtained on the Weizmann complex dataset, the BMCA dataset is a lot more challenging as it is longer and has more complex actions. In fact some of the actions are duals to others in the set; these include the ”stand to sit”, ”sit to stand”, ”stand to lay”, ”lay to stand”, ”stand to

	walk	run	stand to sit	sit to stand	stand to lay	lay to stand	stand to bend	bend to stand	wave1	wave 2	kick
walk	0.91	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09
run	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
stand to sit	0.30	0.00	0.70	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
sit to stand	0.10	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	0.00	0.10
stand to lay	0.50	0.00	0.00	0.00	0.25	0.00	0.08	0.00	0.00	0.00	0.17
lay to stand	0.11	0.00	0.00	0.00	0.00	0.56	0.00	0.11	0.00	0.00	0.22
stand to bend	0.17	0.08	0.00	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.25
bend to stand	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.83	0.00	0.00	0.08
wave1	0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.25	0.00	0.50
wave2	0.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.63	0.25
kick	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.92

Figure 4.8: Confusion matrix for single-view joint segmentation and recognition on BMCA dataset. Average accuracy is **66.74 %**.

bend" and "bend to stand" classes. Without the spatiotemporal constraints, it would be very hard to discriminate between these action duals. With the spatiotemporal constraints, however, there is no misclassification between action duals, as evidenced by the zero off-diagonal values in the confusion matrix.

Next, we evaluate the multi-view DKM algorithm on the same test sets by including the remaining camera views. The multi-view DKMs are evaluated on the test set using the same joint segmentation and recognition strategy used in the single-view case. The resulting confusion matrix is presented in Fig. 4.9. It is clear that the addition of multiple views significantly improves the action detection and recognition performance. Specifically, an average accuracy of **81.28 %** is achieved which represents a **14.54 %** increase in accuracy. We believe that incorporating more overlapping views around the capture volume can improve the accuracy even further, but have not yet tested this hypothesis.

	walk	run	stand to sit	sit to stand	stand to lay	lay to stand	stand to bend	bend to stand	wave1	wave 2	kick
walk	0.89	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.12
run	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
stand to sit	0.00	0.00	0.94	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06
sit to stand	0.11	0.00	0.00	0.83	0.00	0.00	0.00	0.00	0.00	0.00	0.06
stand to lay	0.00	0.00	0.00	0.00	0.72	0.00	0.17	0.11	0.00	0.00	0.00
lay to stand	0.00	0.00	0.00	0.00	0.00	0.88	0.06	0.06	0.00	0.00	0.00
stand to bend	0.06	0.06	0.00	0.00	0.00	0.00	0.72	0.17	0.00	0.00	0.00
bend to stand	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.94	0.00	0.00	0.06
wave1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.79	0.07	0.07
wave2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	0.67	0.00
kick	0.28	0.06	0.00	0.00	0.00	0.00	0.00	0.06	0.06	0.00	0.55

Figure 4.9: Confusion matrix for multi-view joint segmentation and recognition on BMCA dataset. Average accuracy is **81.28 %**.

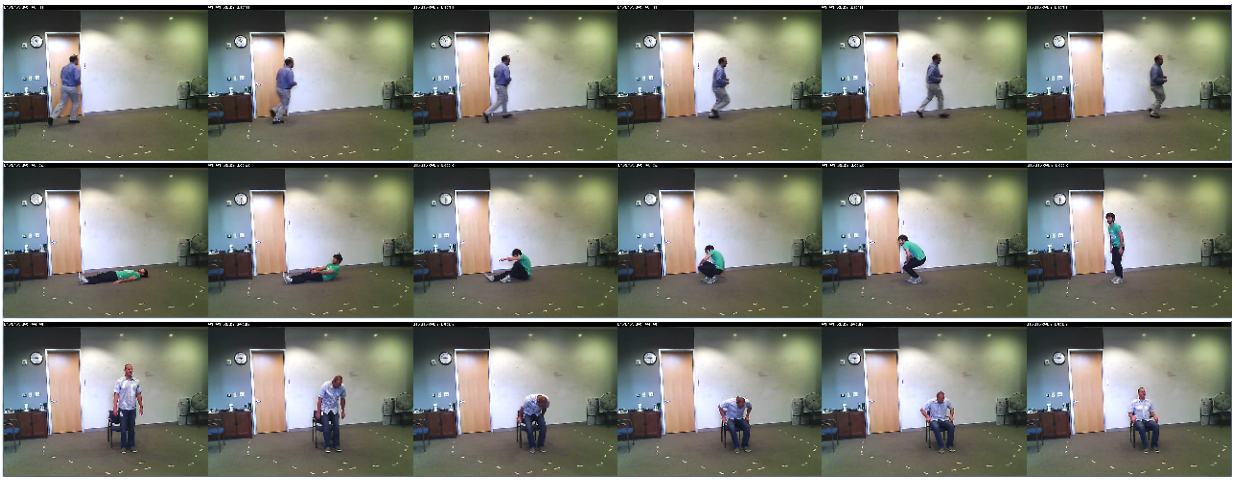


Figure 4.10: Keyframes for a few actions in the BMCA dataset. The first row shows the 6 keyframes corresponding to the action ”run”. The second and third rows show the chosen keyframes for the actions ”lie-to-stand” and ”stand-to-sit” respectively.

4.6 Conclusion

We have presented a framework for the joint detection and recognition of human actions on long, complex video sequences. Our method is well suited for situations where the camera sensors and the base station are connected only by a band-limited communication channel. We have made three primary contributions in this chapter. The first includes a framework for feature extraction on a wireless smart camera that can minimize its power and bandwidth requirements. Our second contribution is the adaptation of the DPM object detection framework for single-view and multi-view action detection in continuous video, and our final contribution is a novel scheme to learn the bias and parameters of our deformable keyframe models. We have experimentally validated our algorithm on a publicly available dataset, and have demonstrated the competitiveness of our approach against state-of-the-art methods. Finally, we have introduced a novel multi-view continuous action data set called the Bosch Multiview Complex Action dataset and extensively validated the performance of our system using this dataset.

Our investigations have led us to several intriguing open problems for future investigation. First, our framework for evaluating multiple deformable keyframe models concurrently on the data by subtracting the model bias may deteriorate when more action classes are introduced. Perhaps a detection strategy similar to the generalized Hough transforms adopted by [10] could make the detectors more robust. Second, our best detection and recognition performance on our dataset is 82%. In order to successfully deploy such a system in real-world surveillance applications, the recognition rates have to be improved dramatically (e.g. $> 99\%$) with minimal false positives. Finally, robust techniques must be studied in order to deal with real world situations such as poor lighting, and occlusions in the scene. In such settings, a smart sensor selection scheme might have to be explored.

Chapter 5

Joint Categorization and Segmentation of Objects

5.1 Introduction

One of the most important tasks for a situation awareness system is to detect people and objects of interest precisely in image streams. In many situations, objects can occur in environments where the background has similar color or texture when compared to some parts of the object. This introduces and extra level of complexity when the image pixels corresponding to the object needs to be segmented and categorized. Several formulations based on Random Fields (RFs) have been proposed for Joint Categorization and Segmentation (JCaS) of objects in images [7, 11, 51]. The RF's sites correspond to pixels or superpixels of an image and one defines potential functions (typically over local neighborhoods) which define costs for the different possible assignments of labels to several different sites. Since the segmentation is unknown a priori, one cannot define potential functions over arbitrarily large neighborhoods as that may cross object boundaries. Categorization algorithms extract a set of interest points from the entire image and solve the categorization problem by optimizing cost functions that depend on the feature descriptors extracted from these interest points. There is some disconnect between segmentation algorithms which consider local neighborhoods and categorization algorithms which consider non-local neighborhoods. In this thesis, we propose to bridge this gap by introducing a novel formulation which uses models of objects with deformable parts, classically used for object categorization, to solve the JCaS problem.

The goal of JCaS is to assign an object category label to each pixel in the image. Several solutions to JCaS use RF-based formulations, wherein algorithms define a RF whose sites correspond to pixels in the image and/or superpixels of the image [7, 11, 15, 36, 38, 37, 55, 53, 58, 75, 77, 54, 86, 93, 99, 106]. To solve the JCaS problem, one defines potential functions (or potentials) which define costs for the different assignments of category labels to the sites. These potentials aggregated over local neighborhoods are then used to define an energy function over the different labelings, the minimizer of which is used to obtain a labeling for the image.

The potential functions used by most of the existing algorithms are local in nature. The unary

potential for a site, which depends on the label of that single site only, is typically defined by using feature descriptors extracted from a local neighborhood of the site, e.g., [86, 53]. The features cannot be extracted from arbitrarily large neighborhoods since they might cross the objects' boundaries. Some methods consider non-local interest regions [106, 93] and use them to define pairwise potentials, which depend on labels of just two sites. Unary and pairwise potentials are typically not sufficient to describe all relationships amongst the sites. Hence, some algorithms use higher order potentials that depend on several sites [53, 54, 55]. While these potentials are also defined over local neighborhoods such as neighboring pixels or superpixels, there are a few exceptions [54, 87].

We argue that one can improve performance by using potentials that are defined over larger non-local neighborhoods, preferably all the regions covered by an object. However, such potentials can lead to a computational bottleneck. Therefore, it is preferable to define potentials over some representative subset region of the object. In this work, we propose to use models of objects with deformable parts [31, 32, 110], which have traditionally been used for object categorization, to define higher order potential functions over non-trivial non-local neighborhoods. These models assume that each object has a set of parts and the problem of detection corresponds to finding the locations of these parts in the image. Our work is motivated by the fact that the locations of the object's parts help define the non-local neighborhoods for our proposed potentials.

Main contributions. We propose to address the aforementioned issues by integrating deformable parts models with RF formulations for JCaS. We assume that we are given a set of hypotheses as the output of detectors based on deformable parts models. Each hypothesis specifies for the object, a size, a pose and the locations for the object's parts. Given this, we propose a new energy function for JCaS with the following properties.

1) The energy function solves for detection and segmentation in a unified framework. The solution obtained by minimizing this function provides (i) a segmentation of the image, (ii) a list of the hypotheses that are accepted from the given ones, and (iii) a list of the visible parts for each of the accepted hypothesis.

2) Our key contribution is the design of two new higher order potential functions for defining the above energy function. The first family of potentials models the detection score for the deformable parts model. The binary-valued variables of this family of potentials indicate whether a part is detected/occluded at a certain location and the potential encodes the object detection score as a function of the visible parts only. The second family of potentials is used to model the *shape prior* of a part. Specifically, a part's shape prior provides for each pixel in the support region of that part, the probability that it belongs to the foreground object. Our proposed potentials use these probabilities to bias the segmentations of the pixels towards the foreground object label.

3) The problem of computing the minimizer of our proposed function is a discrete optimization problem, which can be NP-hard in general. We show that a global optimum to our optimization problem can be computed using min-cut.

Related work. The following are a few examples that have used object models for non-local potentials for JCaS. [58] modeled the object using multiple blobs. [38] and [37] used the output of object detectors to localize the objects in images. [87] and [99] used Bag of Features as the object model, while [7] and [11] used Poselets for their model. Our work, in contrast, uses the deformable

parts model.

The works most closely related to our work are those of [51], [55] and [111]. [51] was perhaps the first work to use deformable parts models for object segmentation. The solution is obtained via an iterative process where the algorithm alternates between sampling from the space of possible hypothesis and computing the segmentation given the hypothesis. [111] extends [51] to deal with multiple object categories. [111] takes as input a set of hypothesis, all of which are used for segmentation. Our proposed framework has a few differences with these. First, we model the detection score as a function of the visible parts, while the above do not. Second, [111] computes the solution using EM, while we compute the segmentation in a single step using min-cut. Finally, in contrast to [111], our algorithm allows for rejection of some of the hypotheses provided as input.

[55] takes as input a set of hypotheses giving the locations of different parts of the image. Given these hypotheses, [55] defines a potential function which penalizes the number of pixels in the support region for each object part, which deviate from the foreground label. There is no shape prior used to bias the pixels differently based on their location in an object part's support region. Moreover, they do not model the detection score as a function of the visible parts.

Outline. In §5.2, we review some definitions that are relevant to our proposed formulation. In §5.3, we propose a new cost function for JCaS. We introduce two new higher order potentials for this cost function and discuss the constraints on these potentials that make them amenable to efficient inference using min-cut. We outline how the parameters of our cost function can be learned using max-margin methods. In §5.4, we evaluate the performance of our formulation on the PARSE dataset [78] and highlight our framework's advantages/limitations.

5.2 Review

In this section, we briefly review some concepts relevant to our formulation.

5.2.1 Random fields (RFs) formulations for JCaS

Given an image I, we define a RF, the set of whose sites is denoted as $\mathcal{V}$. These sites correspond to pixels or superpixels of the image. A binary-valued random variable $X(v_i)$ is defined at each site $v_i \in \mathcal{V}$ and can take any value $x(v_i)$ in the set of possible labels $\mathcal{B} = \{0, 1\}$. Any assignment of labels to the random variables is referred to as a *labeling* and is denoted as $\mathbf{x} \in \mathcal{B}^{|\mathcal{V}|}$. We denote the restriction of the random variables and labeling to a set of sites $A \subseteq \mathcal{V}$ as $\mathbf{X}(A)$ and $\mathbf{x}(A)$, respectively. Note that $x(v_i)$ is the restriction of $\mathbf{x}$ to the site v_i. Though the set of possible labels can contain several values for multiple categories, we restrict our analysis to the case of two labels for the ease of exposition.

The neighborhood of the RF is defined using the set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. An edge that spans two sites v_i and v_j is denoted by $\mathbf{e}_{ij}$. Larger neighborhoods are defined using cliques, where a clique $c \subset \mathcal{V}$ defines a set of sites, e.g., the set of pixels in a superpixel. We denote the set of all cliques in the RF as $\mathcal{C}$. One defines *potential functions* for each clique to model the scores for different assignments of labels to the clique. The following are a few commonly used potentials.

A *unary potential* $\psi_i(x(v_i); I)$ is defined for each site $i \in \mathcal{V}$, such that $\psi_i(b; I)$ defines the cost of assigning the label $b \in \mathcal{B}$ to the site i. This cost is typically computed using appearance-based or location-based feature descriptors. A *pairwise potential* $\psi_{ij}(x(v_i), x(v_j); I)$ is defined for each pair of neighboring sites v_i and $v_j \in \mathcal{V}$, where $\mathbf{e}_{ij} \in \mathcal{E}$, such that $\psi_{ij}(b_i, b_j; I)$ defines the cost of assigning labels b_i and b_j to the sites v_i and v_j, respectively. These potentials help enforce the spatial smoothness of $\mathbf{x}$ and align the edges across which the labeling changes with the edges in the image. They are also used to encode context.

Recent work has addressed the use of higher order potentials defined on larger cliques [53, 54, 55, 87]. A *higher order potential* $\psi_c(\mathbf{x}(c); I)$ is defined on the clique $c \in \mathcal{C}$, such that $\psi_c(\mathbf{b}_c; I)$ is the cost of assigning the labels $\mathbf{b}_c \in \mathcal{B}^{|c|}$ to the clique c. The potential $\psi_c(\mathbf{x}(c); I)$ can be defined over the the clique of pixels that belong to a superpixel. It can also be used to encode higher order contextual information about co-occurence of different categories [54] or to encode bin counts of histograms of quantized descriptors of interest points [87].

Most algorithms solve JCaS by minimizing an energy function of the form

$$E_1(\mathbf{x}; I) = \sum_{c \in \mathcal{C}} \lambda_c \psi_c(\mathbf{x}(c); I), \tag{5.1}$$

where $\forall c \in C, \lambda_c \in \mathbb{R}$. Note that (5.1) includes unary and pairwise potentials as special cases when $|c| = 1$ and 2, respectively. $E_1(\mathbf{x}; I)$ is typically designed such that min-cut based solvers provide the global minimum for the 2-label case and a local minimum (with optimality bounds) for the multi-label case.

As described in §5.1,it is preferable to have global object models that consider larger non-local neighborhoods, preferably all the sites with the same label. Such neighborhoods cannot be imposed apriori because the labeling is unknown.

5.2.2 Detection of objects with deformable parts

The algorithms in this genre assume that an object consists of $P \in \mathbb{Z}_+$ parts [31, 30, 32, 110]. Given an image I, a hypothesis θ specifies the object's pose $\pi(\theta)$, object's scale (size) $s(\theta)$ and a set of locations $l(\theta) = [l_1(\theta), \ldots, l_P(\theta)]^\top \in \Omega(I)^P$ for the different parts, where $\Omega(I) \in \mathbb{R}^2_+$ denotes the pixel domain of image I. The algorithms compute a detection score for the different hypotheses. The hypotheses with scores better than a threshold (say κ) are treated as accepted.

To define a detection score for a hypothesis θ, the algorithms consider two different kinds of cost functions. The first type of cost function is an appearance-based cost for each of the different parts. For the p^{th} part ($p = 1, \ldots, P$), one extracts feature descriptors from a support region (say $R_p(\theta)$) around $l_p(\theta)$, where the size of the support region depends on the object's pose, scale and the part. The appearance-based cost $\phi_p^{\text{app}}(\theta; I)$ for the p^{th} part is then computed as the output of a linear filter applied to these features descriptors, where the filter's coefficients depend on the pose, scale and part.

The second cost function takes into account the constraints on the relative locations of the different parts. Given locations $l_{p_1}(\theta)$ and $l_{p_2}(\theta)$ for parts p_1 and p_2, the cost $\phi^{\text{def}}_{p_1,p_2}(\theta)$ is a quadratic function of the entries of the vector $l_{p_1}(\theta) - l_{p_2}(\theta)$, where the coefficients of the quadratic function

depend on the object's pose and scale. While one may construct a deformation cost for each of the $\frac{P(P-1)}{2}$ possible pairs of parts, most algorithms assume, for computational ease, that only a subset of these pairs are relevant for the purpose of detection. In fact, it is assumed that this subset of pairs can be represented using a tree. These connections between the parts are defined by the set of edges $\mathcal{E}_{\text{obj}} = (p_1, p_2)$.

Given an image I, one then defines the detection score for a candidate θ as

$$E_2(\theta; I) = \sum_{p=1}^{P} \phi_p^{\text{app}}(\theta; I) + \sum_{(p_1,p_2)\in\mathcal{E}_{\text{obj}}} \phi_{p_1,p_2}^{\text{def}}(\theta) \tag{5.2}$$

Due to the small number (say M) of possible poses and number (say S) of scales considered by detection algorithms, it is possible to do a brute force computation of the energy for the different poses and scales. Since the number of possible locations of the parts is high, i.e., $|\Omega(I)|^P$, it is not possible to compute the energy for all possible locations. To address this, given partial information for a hypothesis θ in terms of the pose $\pi(\theta)$ and scale $s(\theta)$, the part locations $l(\theta)$ that minimize $E_2(\theta; I)$ can be found using dynamic programming with time complexity $O(|\Omega(I)|P)$ [30]. We note that in the literature, the best hypothesis is typically obtained by solving a maximization problem. We can always reformulate the problem to get an equivalent minimization problem. In our work, we assume, without loss of generality, that the best hypotheses are obtained by solving a minimization problem, i.e., lower hypothesis scores are considered better.

In this formulation, the algorithm assumes that for a given pose, each part is assumed to be detected/visible in the image. There has been work to deal with occlusions, but an occluded pose is modeled as a pose different from the original pose [31]. We argue that it is of interest to explicitly model occlusion of parts within a certain pose rather than modeling occlusions using different poses.

5.3 A Novel Energy Function for JCaS

In this section, we define a new energy function to model non-local interactions amongst the sites of RFs for JCaS. For expositional ease, we make two simplifying assumptions. First, the number of parts (say P) is the same for all the poses. This is not necessary in practice. Second, we assume that the image is segmented into two groups only – an object of a particular category vs. background. Our analysis can be extended to deal with multiple semantic categories too.

We now introduce some notation. Given an image I, we denote the set of sites representing the pixels as $\mathcal{V}_{\text{pixels}} = \{v_1, \ldots, v_N\}$, where $N = |\Omega(I)|$. We do not introduce any additional sites for superpixels since potentials defined over superpixels can be redefined as potentials defined over the pixels [53]. We assume that we are given a set of H hypotheses, $\Theta = \{\theta_1, \ldots, \theta_H\}$. For each hypothesis θ_h (where $h = 1, \ldots, H$), we define a set of $P + 1$ sites $\mathcal{V}_{\text{obj}}(\theta_h) = \{v_0^h, v_1^h, \ldots, v_P^h\}$. The site v_0^h is used to represent the h^{th} hypothesis θ_h and for $p = 1, \ldots, P$, the site v_p^h is used to represent the p^{th} object part for hypothesis θ_h.

We introduce binary-valued variables for the sites. For each site $v_i \in \mathcal{V}_{\text{pixels}}$, the variable $x(v_i)$ takes value 0 or 1 and represents segmentation as background or object, respectively. For each site

v_0^h, the variable $x(v_0^h)$ takes value 0 or 1 and represents whether the hypothesis θ_h is rejected or accepted, respectively. For each site $v_p^h \in \mathcal{V}_{\text{obj}}(\theta_h)$, where $p > 0$, $x(v_p^h)$ takes value 0 or 1 and represents whether the p^{th} part for the h^{th} hypothesis is occluded or visible, respectively.

In this work, we propose to solve the JCaS problem by computing the values for the variables $\mathbf{x}$ that minimize an energy function of the form

$$\begin{aligned} E^{\text{JCaS}}(\mathbf{x}; I, \Theta) = \lambda^{\text{seg}} E^{\text{seg}}(\mathbf{x}(\mathcal{V}_{\text{pixels}}); I) + \lambda^{\text{hyp}} \sum_{h=1}^{H} E^{\text{det}}(\mathbf{x}(\mathcal{V}_{\text{obj}}(h)); I, \theta_h) \\ + \sum_{h=1}^{H} \sum_{p=1}^{P} \lambda_p^{\text{shape}}(\pi(\theta_h)) E_p^{\text{shape}}(x(v_p^h), \mathbf{x}(R_p(\theta_h)); I, \theta_h), \end{aligned} \tag{5.3}$$

where λ^{seg}, λ^{hyp} and $\lambda_p^{\text{shape}}(\cdot)$ are all non-negative scalars, and $R_p(\theta_h)$ is the support region for the p'th part in hypothesis θ_h. The term $E^{\text{seg}}(\cdot)$ is a segmentation-based energy function. It encodes the cost of assigning segmentation labels to the pixels, where the cost is computed using feature descriptors such as color, texture, etc. This energy can also be thought of in more general terms and can be replaced by energy functions used by existing JCaS algorithms.

Our main contribution is the design of the energies $E^{\text{det}}(\cdot)$ and $E_p^{\text{shape}}(\cdot)$. The term $E^{\text{det}}(\cdot)$ is a detection-based energy function and computes the detection score for each of the H hypotheses, as a function of the visible parts only. The third term $E_p^{\text{shape}}(\cdot)$ is an energy function that connects the segmentation and detection terms. It helps encode how the p^{th} part, if visible, affects the segmentation of the image region where the part is detected. We will discuss later, that it also helps in the use of the segmentation of an image region to verify whether a part is visible or not. In what follows, we define these energy functions and discuss how we can obtain $\mathbf{x}$ as the minimizer of $E^{\text{JCaS}}(\cdot)$.

5.3.1 Definition of the energy terms

5.3.1.1 Detection.

We first extend the detection score defined in (5.2) by introducing the binary-valued variables $x(v_p^h)$ that model the visibility/occlusion of the parts, as

$$\phi(\mathbf{x}; I, \theta_h) = \sum_{p=1}^{P} \phi_p^{\text{app}}(\theta_h; I) x(v_p^h) + \sum_{(p_1,p_2)\in\mathcal{E}_{\text{obj}}} \phi_{p_1,p_2}^{\text{def}}(\theta_h) x(v_{p_1}^h) x(v_{p_2}^h). \tag{5.4}$$

The appearance score for the p^{th} part is accounted for only if it is visible ($x(v_p^h) = 1$). The deformation score for a pair of parts is accounted for only when both parts are visible. Hence, the detection score depends on the visible parts only.

Given this definition of the score, we define the hypothesis score as follows

$$E^{\text{det}}(\mathbf{x}(\mathcal{V}_{\text{obj}}(\theta_h)); I, \theta_h) = \begin{cases} \phi(\mathbf{x}; I, \theta_h) - \kappa & \text{if } \phi(\mathbf{x}; I, \theta_h) \le \kappa \\ 0 & \text{if } \phi(\mathbf{x}; I, \theta_h) \ge \kappa \text{ and } \mathbf{x}(\mathcal{V}_{\text{obj}}(\theta_h)) = \mathbf{0}, \\ \infty & \text{if } \phi(\mathbf{x}; I, \theta_h) \ge \kappa \text{ and } \mathbf{x}(\mathcal{V}_{\text{obj}}(\theta_h)) \neq \mathbf{0} \end{cases} \tag{5.5}$$

where κ is a pre-defined threshold applied to the detection score, to accept a hypothesis (recall from §5.2.2). We have considered three different cases in defining $E^{\text{det}}(\cdot)$. In the first case, the detection score is below the threshold and the hypothesis is accepted. The cost paid in this case is a negative value and is precisely equal to $\phi(\mathbf{x}; I, \theta_h) - \kappa$. When the detection score is above the threshold, we want to reject the hypothesis and we don't want any of the parts to be detected. The third case in (5.5) ensures that none of the parts are detected when the detection score is more than κ, by assigning a very high cost, i.e., ∞, to this undesirable case. The second case in (5.5) corresponds to the case when we reject the hypothesis and no part is detected. In this case, we pay a constant cost 0.

5.3.1.2 Shape prior.

$E^{\text{seg}}(\cdot)$ and $E^{\text{det}}(\cdot)$ are defined on disjoint sets of vertices, i.e., $\mathcal{V}_{\text{pixels}}$ and $\mathcal{V}_{\text{obj}}(\cdot)$, respectively. $E_p^{\text{shape}}(\cdot)$ serves to connect the segmentation variables with the hypotheses variables. Given a hypothesis θ_h, we define for each part p, a shape prior (see Figure 5.1a) over its support region $R_p(\theta_h)$, as

$$\forall v_i \in R_p(\theta_h) : \xi(v_i, p) = prob(x(v_i) = 1 | x(v_p^h) = 1) \tag{5.6}$$

More specifically, the shape prior specifies for each pixel in the support region of a visible part, the probability that it will be assigned to the foreground object. Now, note that it is straightforward to define an energy function for the p^{th} part, as

$$\sum_{v_i \in R_p(\theta_h)} \left(-\xi(v_i, p) x(v_i) x(v_p^h) \right). \tag{5.7}$$

Specifically, when the p^{th} part is detected ($x(v_p^h) = 1$) and a pixel v_i in its support region is assigned to the foreground, a negative cost $-\xi(v_i, p)$ is paid. This implies that all the pixels which have a high probability (as given by the shape prior) of belonging to the foreground, will have a greater bias towards being segmented as foreground. In this manner, we see how the detection of parts can help improve the segmentation. However, there must be a symbiotic interplay between segmentation and detection, and we argue that segmentation must also help improve the detection. We propose a constraint that a part must be treated as being detected/visible, only if a sufficient number of pixels in its support region are segmented as belonging to the foreground. To this effect, we define the energy function as

$$E_p^{\text{shape}}(x(v_p^h), \mathbf{x}(R_p(\theta_h); I, \theta_h), = \begin{cases} \sum_{v_i \in R_p(\theta_h)} -\xi(v_i, p) \beta_p(\pi(\theta_h)) & \text{if } x(v_p^h) = 0 \\ \sum_{v_i \in R_p(\theta_h)} -\xi(v_i, p) x(v_i) & \text{if } x(v_p^h) = 1 \end{cases}, \tag{5.8}$$

where $\beta_p(\pi(\theta_h)) > 0$. When the p^{th} part is not detected, a constant cost (which does not depend on the segmentation) is paid. When the part is detected, the cost depends on the segmentation in the support region. Moreover, notice from (5.8) that when a sufficient number of pixels in $R_p(\theta_h)$ are assigned to the foreground, i.e., when $\sum_{v_i \in R_p(\theta_h)} -\xi(v_i, p) x(v_i) \leq \beta_p(\pi(\theta_h)) \sum_{v_i \in R_p(\theta_h)} -\xi(v_i, p)$, this energy function biases the p^{th} part towards being detected.

5.3.2 Inference

The potentials defined in (5.5) and (5.8) are higher order potentials that depend on the labels of more than two sites. We will now show how these potentials can be expressed using unary and pairwise potentials. Since energy functions with unary and pairwise potentials can be minimized using min-cut, our proposed potentials can be integrated into existing JCaS algorithms that use min-cut based solvers.

To find the global optimum using min-cut, there are no constraints on the unary potentials. The pairwise potentials, however, do need to satisfy the *submodularity* constraint [50]. If one considers pairwise potentials that are defined over two binary-valued variables, say y_1 and y_2, the pairwise potentials $\gamma_1 y_1 y_2$ and $\gamma_2 \bar{y}_1 y_2$ (where $\bar{y}_1 = 1 - y_1$) are submodular only if $\gamma_1 \leq 0$ and $\gamma_2 \geq 0$ [50].

We first see that the energy $E_p^{\text{shape}}(\cdot)$ defined in (5.8) can be rewritten as

$$E_p^{\text{shape}}(x(v_p^h), \mathbf{x}(R_p(\theta_h); I, \theta_h) = \sum_{v_i \in R_p(\theta_h)} -\xi(v_i, p) \left(\beta_p(\pi(\theta_h))\bar{x}(v_p^h) + x(v_i)x(v_p^h)\right). \tag{5.9}$$

It is easy to verify that for $x(v_p^h) = 0$ and $x(v_p^h) = 1$, the score in (5.9) is exactly the same as that in (5.8). The first term in the summation in (5.9) is a unary term that depends only on $x(v_p^h)$. The second term is a pairwise potential that depends on $x(v_i)$ and $x(v_p^h)$. In this case, we note that by its definition in (5.6), $\xi(v_i, p) \geq 0$. Therefore, the potential $-\xi(v_i, p)x(v_i)x(v_p^h)$ is submodular by construction.

We now use the variable $x(v_0^h)$ to rewrite $E^{\text{det}}(\cdot)$ defined in (5.5), as

$$E^{\text{det}}(\mathbf{x}(\mathcal{V}_{\text{obj}}(\theta_h)); I, \theta_h) = (\phi(\mathbf{x}; I, \theta_h) - \kappa)\, x(v_0^h) + \infty \sum_{p=1}^{P} (\bar{x}(v_0^h)x(v_p^h)). \tag{5.10}$$

When $x(v_0^h) = 1$, the hypothesis is accepted and the cost is equal to $\phi(\mathbf{x}; I, \theta_h) - \kappa$. When $x(v_0^h) = 0$, the hypothesis is rejected and the third term $\infty(\bar{x}(v_0^h)x(v_p^h))$ ensures that all the $x(v_p^h) = 0$ when $x(v_0^h) = 0$. The cost paid when $\mathbf{x}(\mathcal{V}_{\text{obj}}(\theta_h)) = \mathbf{0}$ is equal to 0. Therefore, this energy represents the detection energy in (5.5). Now, given the expression in (5.4), we can rewrite the right hand side of (5.10) as

$$\begin{aligned}(\phi(\mathbf{x}; I, \theta_h) - \kappa)x(v_0^h) + \infty \sum_{p=1}^{P} (\bar{x}(v_0^h)x(v_p^h)) &= \sum_{p=1}^{P} \phi_p^{\text{app}}(\theta_h; I)x(v_p^h)x(v_0^h) \\ + \sum_{(p_1,p_2)\in\mathcal{E}_{\text{obj}}} \phi_{p_1,p_2}^{\text{def}}(\theta_h)x(v_{p_1}^h)x(v_{p_2}^h)x(v_0^h) - \kappa x(v_0^h) &+ \infty \sum_{p=1}^{P} (\bar{x}(v_0^h)x(v_p^h)).\end{aligned} \tag{5.11}$$

Notice that the first and second expressions are potentials defined over two variables $\left(x(v_p^h)x(v_0^h)\right)$ and three variables $\left(x(v_{p_1}^h)x(v_{p_2}^h)x(v_0^h)\right)$, respectively. However any solution $\mathbf{x}^*$ that minimizes the energy satisfies the constraint that $\forall p = 1, \ldots, P, x^*(v_p^h) = 1$, only if $x^*(v_0^h) = 1$. To this

effect, $x^*(v_p^h)x^*(v_0^h) = 1$, only if $x^*(v_p^h) = 1$. Similarly, $x^*(v_{p_1}^h)x^*(v_{p_2}^h)x^*(v_0^h) = 1$, only if $x^*(v_{p_1}^h)x^*(v_{p_2}^h) = 1$. Hence, we can drop $x(v_0^h)$ in the first and second terms and rewrite $E^{\text{det}}(\cdot)$ as

$$\begin{aligned} E^{\text{det}}(\mathbf{x}(\mathcal{V}_{\text{obj}}(\theta_h)); I, \theta_h) = \sum_{p=1}^{P} \phi_p^{\text{app}}(\theta_h; I) x(v_p^h) + \sum_{(p_1,p_2)\in\mathcal{E}_{\text{obj}}} \phi_{p_1,p_2}^{\text{def}}(\theta_h) x(v_{p_1}^h) x(v_{p_2}^h) \\ - \kappa x(v_0^h) + \infty \sum_{p=1}^{P} (\bar{x}(v_0^h) x(v_p^h)), \end{aligned} \tag{5.12}$$

such that the minimizers of (5.10) and (5.12) are the same. The first and third terms in (5.12) are unary potentials. The fourth term $\infty\bar{x}(v_0^h)x(v_p^h)$ is submodular by construction. The second term $\phi_{p_1,p_2}^{\text{def}}(\theta_h)x(v_{p_1}^h)x(v_{p_2}^h)$ is submodular if and only if $\phi_{p_1,p_2}^{\text{def}}(\theta_h) \leq 0$. This score $\phi_{p_1,p_2}^{\text{def}}(\theta_h)$ is a quadratic function computed as

$$\phi_{p_1,p_2}^{\text{def}}(\theta_h) = \begin{bmatrix} dl_1 \\ dl_2 \\ 1 \end{bmatrix}^\top \begin{bmatrix} a_1(s(\theta_h),\pi(\theta_h)) & 0 & b_1(s(\theta_h),\pi(\theta_h)) \\ 0 & a_2(s(\theta_h),\pi(\theta_h)) & b_2(s(\theta_h),\pi(\theta_h)) \\ b_1(s(\theta_h),\pi(\theta_h)) & b_2(s(\theta_h),\pi(\theta_h)) & c(s(\theta_h),\pi(\theta_h)) \end{bmatrix} \begin{bmatrix} dl_1 \\ dl_2 \\ 1 \end{bmatrix}, \tag{5.13}$$

where $[dl_1, dl_2]^\top = l_{p_1}(\theta_h) - l_{p_2}(\theta_h)$ [30]. Note that by definition, $a_1(s(\theta_h),\pi(\theta_h)) > 0$ and $a_2(s(\theta_h),\pi(\theta_h)) > 0$ [30]. We now describe how the parameters of $\phi_{p_1,p_2}^{\text{def}}(\cdot)$ can be updated for a given image, such that the classification results are not affected and $\phi_{p_1,p_2}^{\text{def}}(\theta_h) \leq 0$ for all possible $(l_{p_1}(\theta_h), l_{p_2}(\theta_h))$.

If we update $\phi_{p_1,p_2}^{\text{def}}(\cdot)$ to $\tilde{\phi}_{p_1,p_2}^{\text{def}}(\cdot)$, such that all the parameters are kept constant but $c(\cdot)$ is updated as $\tilde{c}(s(\theta_h),\pi(\theta_h)) = c(s(\theta_h),\pi(\theta_h)) + \Delta c(s(\theta_h),\pi(\theta_h))$, we have for all θ_h, $\tilde{\phi}_{p_1,p_2}^{\text{def}}(\theta_h) = \phi_{p_1,p_2}^{\text{def}}(\theta_h) + \Delta c(s(\theta_h),\pi(\theta_h))$. This does not alter the relative ordering of the scores of the different hypotheses. The detection results are the same if one updates the threshold κ as $\tilde{\kappa} = \kappa + \Delta c(s(\theta_h),\pi(\theta_h))$.

Given an image I, since there are only a finite number of locations used to compute the expression in (5.13), we can always find a $\Delta c(s(\theta_h),\pi(\theta_h))$ for that image, such that $\tilde{\phi}_{p_1,p_2}^{\text{def}}(\theta_h) \leq 0$ for all possible (l_{p_1}, l_{p_2}). This implies that we can always update the parameters to construct submodular pairwise potentials.

5.3.3 Parameter learning

Notice that the energy $E^{\text{JCaS}}(\mathbf{x}; I, \Theta)$ defined in (5.3), can be rewritten as

$$\begin{aligned} E^{\text{JCaS}}(\mathbf{x}; I, \Theta) &= \mathbf{w}^\top \Psi(\mathbf{x}; I, \Theta) \\ &= \begin{bmatrix} \lambda^{\text{seg}} \\ \lambda^{\text{hyp}} \\ \vdots \\ \lambda_p^{\text{shape}}(\pi_m) \\ \vdots \end{bmatrix}^\top \begin{bmatrix} E^{\text{seg}}(\mathbf{x}(\mathcal{V}_{\text{pixels}}); I) \\ \sum_{h=1}^{H} E^{\text{det}}(\mathbf{x}(\mathcal{V}_{\text{obj}}(h)); I, \Theta) \\ \vdots \\ \sum_{h=1}^{H} \delta(\pi(\theta_h) = \pi_m) E_p^{\text{shape}}(x(v_p^h), \mathbf{x}(R_p(\theta_h)); I, \theta_h) \\ \vdots \end{bmatrix}, \end{aligned} \tag{5.14}$$

where $\delta(\cdot)$ is the 0-1 indicator function and $\mathbf{w} \in \mathbb{R}^{2+(P\times M)}$ contains the parameters that regulate the relative contributions of the different potentials.

Recall that we segment an image I by minimizing $E(\mathbf{x}; I, \Theta)$. Hence, we want that the true segmentation $\mathbf{y}$ of image I minimize the energy $E(\mathbf{x}; I)$ as $\forall \mathbf{x} \in \mathcal{B}^{N+(H\times(P+1))} \setminus \mathbf{y}, E(\mathbf{x}; I, \Theta) > E(\mathbf{y}; I, \Theta)$, i.e., $\mathbf{w}^\top \Psi(\mathbf{x}; I, \Theta) > \mathbf{w}^\top \Psi(\mathbf{y}; I, \Theta)$. We now describe an optimization problem to learn $\mathbf{w}$, motivated by this property.

Assume that we are given a training set of T images $\{I_t\}_{t=1}^T$ with ground truth labelings $\{\mathbf{y}_t\}_{t=1}^T$. We refer to any labeling of an image that is different from $\mathbf{y}_t$ as a negative example of segmentation. We denote the set of negative examples of segmentations for an image I_t as $\mathcal{U}_t^-$. Since all negative segmentation examples should not be treated equally, we propose to enforce the constraint

$$\forall \mathbf{x} \in \mathcal{U}_t^- : \mathbf{w}^\top \big(\Psi(\mathbf{x}; I_t, \Theta_t) - \Psi(\mathbf{y}_t; I_t, \Theta_t)\big) > \ell(\mathbf{x}, \mathbf{y}_t), \tag{5.15}$$

where $\ell(\mathbf{x}, \mathbf{y}_t)$ is a loss function that quantifies errors in the segmentation, as

$$\ell(\mathbf{x}, \mathbf{y}_t) = \frac{\sum_{v_i \in \mathcal{V}_{\text{pixels}}} y_t(v_i)\bar{x}(v_i)}{\sum_{v_i \in \mathcal{V}_{\text{pixels}}} y_t(v_i)} + \frac{\sum_{v_i \in \mathcal{V}_{\text{pixels}}} \bar{y}_t(v_i)x(v_i)}{\sum_{v_i \in \mathcal{V}_{\text{pixels}}} \bar{y}_t(v_i)}. \tag{5.16}$$

$\ell(\mathbf{x}, \mathbf{y}_t)$ computes the sum of fractions of misclassified sites per category.

Given a regularization parameter $C > 0$, we propose to learn $\mathbf{w}$ by solving

$$\begin{aligned} &\{\mathbf{w}^*, \{\eta_t^*\}_{t=1}^T\} = \underset{\mathbf{w}, \{\xi_t\}_{i=1}^N}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{T}\sum_{t=1}^T \eta_t, \text{ subject to } \forall t = 1, \ldots, T \\ &\text{(a) } \forall \mathbf{x} \in \mathcal{U}_t^- : \mathbf{w}^\top \big(\Psi(\mathbf{x}; I_t, \Theta_t) - \Psi(\mathbf{y}_t; I_t, \Theta_t)\big) \geq \ell(\mathbf{x}, \mathbf{y}_t, \Theta_t) - \eta_t, \\ &\text{(b) } \eta_t \geq 0 \quad \text{and} \quad \text{(c) } \mathbf{w} \geq \mathbf{0}. \end{aligned} \tag{5.17}$$

This formulation is mostly based on [95] and we solve (5.17) using the cutting-plane algorithm described in [95]. While we refer the readers to [95] for the details, we now provide some intuition for (5.17). The constraint (a) is similar to (5.15) except for the non-negative valued slack variable η_t which allow for the violation of (5.15). Constraint (c) ensures that the resulting energy is submodular.

5.4 Experiments

Description of dataset. For the evaluation, we use the Image Parse Dataset [78] which consists of 305 articulated full-body images of people. The first 100 images are used as training data and the remaining 205 as test data. We have manually segmented the images in the dataset for our quantitative evaluation.

Algorithms compared in the evaluation. We first describe the construction of the energy $E^{\text{JCaS}}(\cdot)$. We define $E^{\text{det}}(\cdot)$ using the outputs of the detector of [110]. Although our method can handle multiple detection hypotheses, we use only the highest scoring hypothesis for each image in our

(a) (b)

Figure 5.1: (a) Shape priors generated for 4 part types using the parts model of [110]. (b) Two examples of shape priors being superimposed to generate foreground hypothesis.

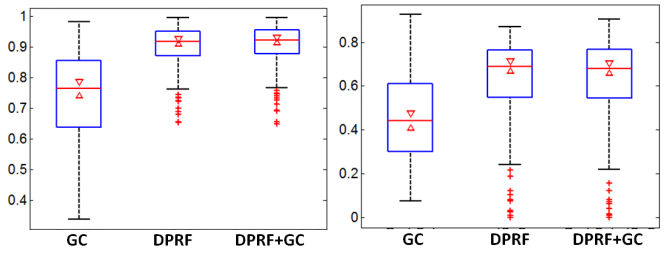
experiments. We now describe how we construct the shape priors for $E_p^{\text{shape}}(\cdot)$. We run the detection algorithm of [110] on the training images. For each part type detected in an image, we find the associated patch in its ground truth segmentation enclosed by the detection box. Averaging the segmentation patches over all the training images provides shape priors similar to those shown in Figure 5.1a. Figure 5.1b shows examples where the learned shape priors are placed at part detection sites. It is clear from this image how the shape priors influence the segmentation of the people.

To construct $E^{\text{seg}}(\cdot)$, we use the given hypothesis to create a color-based unary potential. We fit a a Gaussian Mixture Model (GMM) with 5 components to the RGB-colors of all the image's pixels that lie outside the detection boxes for the parts. Given the color of a pixel v_i in the image, we use this GMM to define the background unary potential $\psi_i^{clr}(0; I)$. We set the foreground unary potentials to zero, i.e., $\psi_i^{clr}(1; I) = 0$. This reduces dependency on color for segmenting the foreground while relying entirely on the detection and shape prior potentials. We also define a color-based pairwise potential as $\psi_{ij}^{clr}(x(v_i), x(v_j)) = \delta(x(v_i) \neq x(v_j))e^{-\beta\|\mathbf{z}(v_i)-\mathbf{z}(v_j)\|^2}$ where $\mathbf{z}(v_i)$ is the RGB color at pixel v_i, and each v_j is in the 4-neighborhood of pixel v_i. In all our experiments we set $\beta = 10$.

As a baseline for comparison, we consider the GrabCut algorithm [80], which considers only unary and pairwise potentials. It alternates between (a) fitting GMMs of color for the foreground/background, given the segmentation, and (b) computing the segmentation, given the potentials constructed with these GMMs. We initialize GrabCut with a segmentation, where we label all the pixels inside the detection boxes for the parts as the foreground, and the rest as background. We run 10 iterations of GrabCut. Unlike the traditional GrabCut, we cannot place any hard constraints on the pixels' labels, since the detection boxes contain pixels belonging to the background as well as foreground. We choose this baseline to show that even if one is given a good initial object detection, using low-level features such as color need not produce good JCaS results. This motivates our argument for object models defined over non-trivial non-local neighborhoods.

We also consider a third algorithm, where we combine our algorithm with GrabCut. We alternate between (a) computing the segmentation by minimizing $E^{\text{JCaS}}(\cdot)$, and (b) improving the color models given the segmentation.

	Foreground		Background	
Method	mean	std	mean	std
GC	0.74	0.15	0.46	0.20
DPRF	0.90	0.07	**0.63**	**0.19**
DPRF +GC	**0.91**	**0.06**	**0.63**	**0.19**



(a) Foreground I/U (b) Background I/U

Figure 5.2: Comparison of I/U for the segmentation results produced by the 3 methods.

The parameters for all the three algorithms are learnt as described in §5.3.3. In what follows, we refer to our method as DPRF (deformable parts + random fields) and to GrabCut as GC. The third algorithm is referred to as DPRF+GC.

Evaluation. We evaluate the segmentations using the Intersection/Union (I/U) metric given by $\frac{\#TP}{\#TP+\#FP+\#FN}$, where TP = true positives, FP = false positives and FN = false negatives. Better segmentation corresponds to higher I/U.

The results are presented in the table and the boxplots in Fig. 5.4. The top/ bottom edge of each boxplot for a set of values indicates the maximum/minimum of the values. The bottom/top extents of the box mark the $25/75$ percentile. The red line in the box indicates the median and the red crosses outside the boxes show potential outliers. The 5% confidence intervals for determining statistical significance of difference between the medians are shown as red triangles.

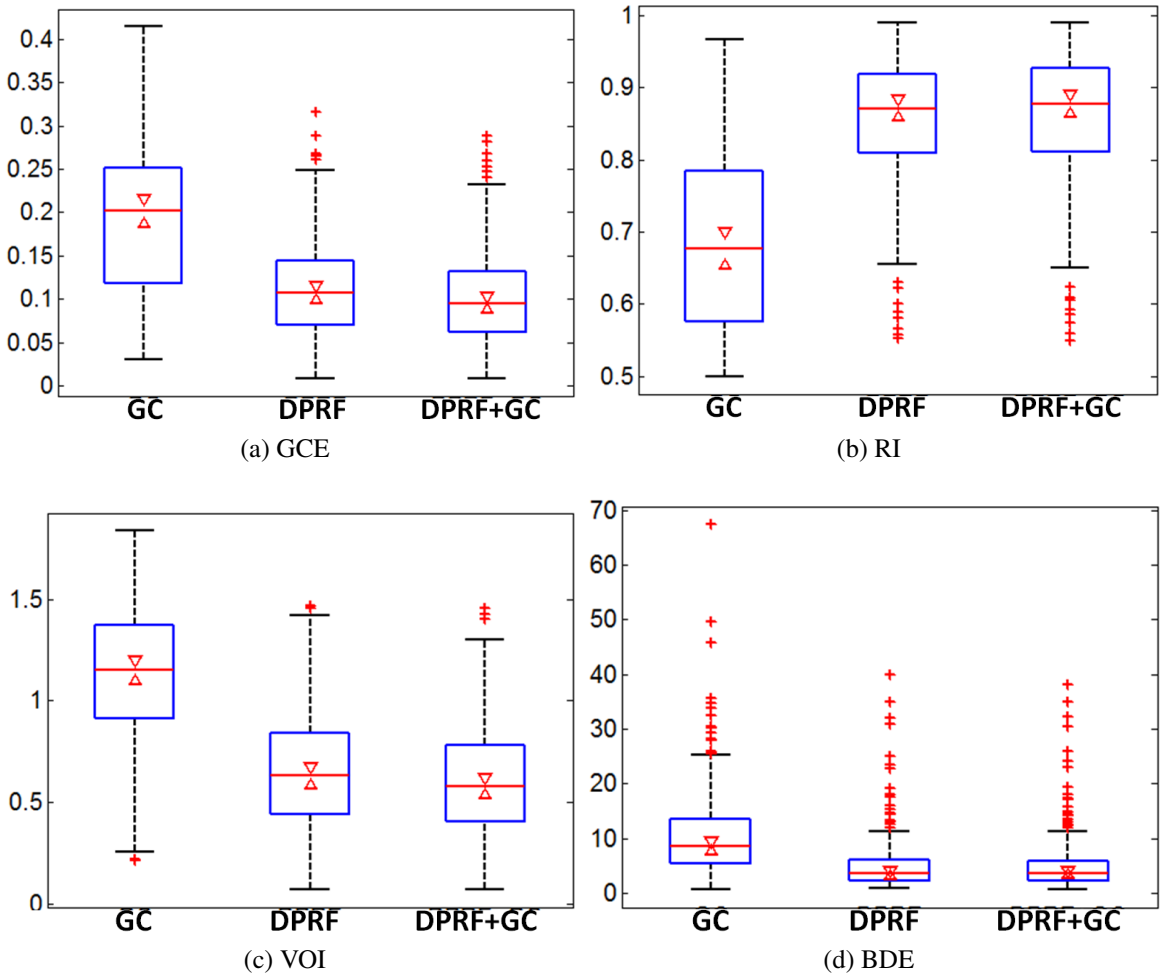
The median I/U is notably lower for GC in comparison to DPRF and the 5% confidence intervals for these results do not overlap. However, the medians for DPRF and DPRF+GC are very similar and the 5% confidence intervals do overlap. This combined with the results in the table help us conclude that (a) DPRF produces better results than GC, and (b) the introduction of color information into DPRF, i.e., DPRF+GC does not produce any significant improvement.

Figure 5.3 presents a qualitative comparison of the results. The first column shows the hypothesis for the deformable parts (from [110]) overlaid on the image. The second column shows the result of pruning some of the detections using DPRF. The third, fourth and fifth columns of the figure show the segmentation produced by GC, DPRF and DPRF+GC, respectively. The first 3 rows show examples where the results of GC are inferior to those produced by DPRF and DPRF+GC. In these examples, the detection algorithm has fit the articulated models to the data reasonably well. The next two examples show scenarios where the detection algorithm errs and detects an extra limb (circled in white). As seen in the second column, DPRF prunes out these errors and provides a better segmentation than GC. The last two rows show examples where GC performs comparable to or outperforms DPRF. The last failure case is due to the poor part detection as seen in the first column in the last row. More success and failure cases are presented in figures 5.5 and 5.6 respectively.

Figure 5.3: Column 1 shows the articulated model overlaid on the images. Column 2 shows the pruned model that has rejected some part detections using DPRF. Columns 3-5 show the segmentations given by GC, DPRF and DPRF+GC. See text for explanation.

5.5 Conclusion

We presented a JCaS framework where we proposed two new families of potentials that combine detection hypothesis with the segmentation of the image. These potentials can be integrated with existing RF-based JCaS algorithms. Results show that the detection hypothesis helps provide good segmentation results, and the segmentation can be used to prune some errors in the hypothesis.



(a) GCE (b) RI

(c) VOI (d) BDE

Figure 5.4: Quantitative comparison of segmentation produced by the 3 different methods using (a) Global Consistency error (GCE), (b) Rand Index (RI), (c) Variation of Information (VOI) and (d) Boundary Displacement Error (BDE). Note that better segmentation quality corresponds to a lower GCE, lower VOI, lower BDE and higher RI.

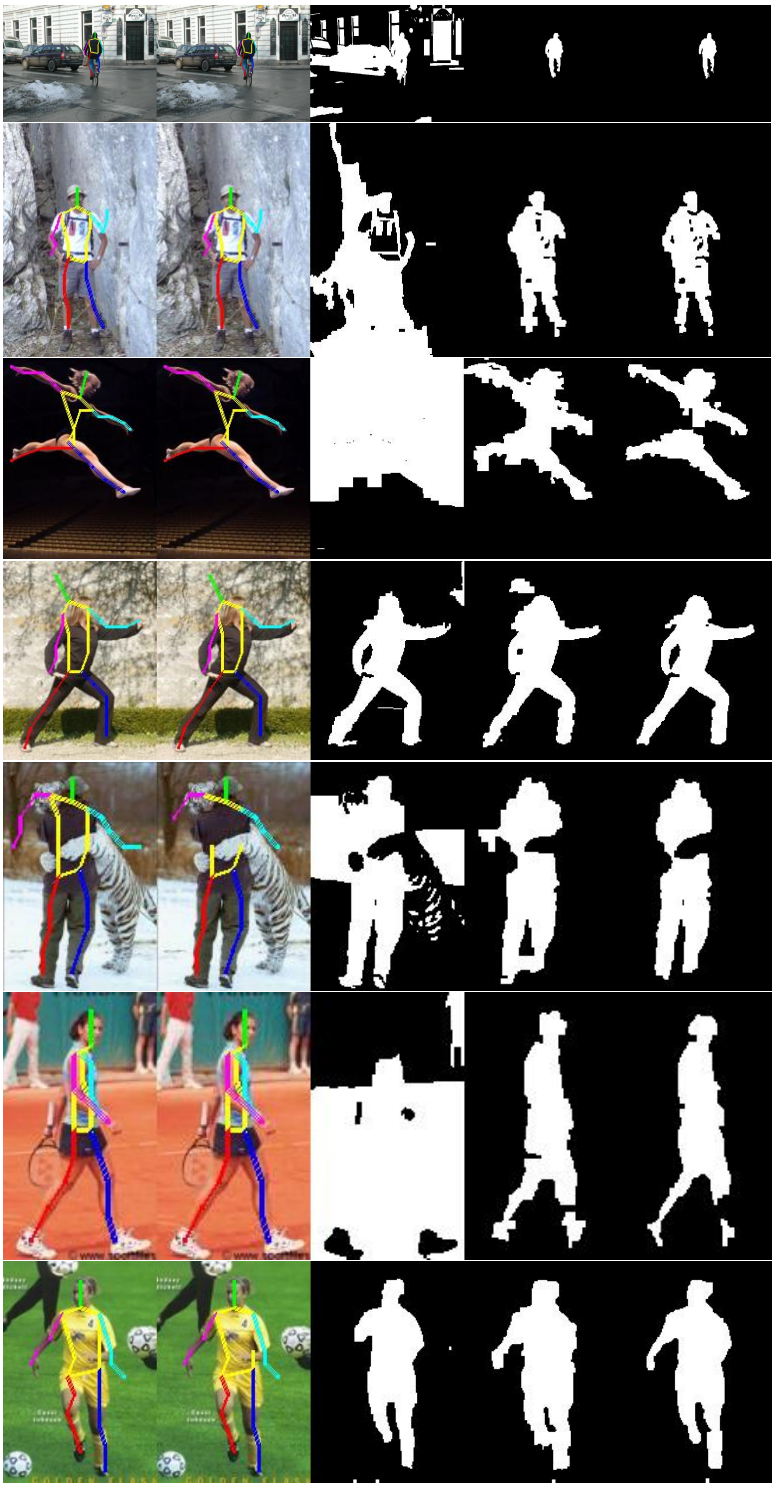


Figure 5.5: **Success cases**- Column 1 shows the articulated model overlaid on the images. Column-2 shows the pruned model that has rejected some part detections using DPRF. Columns 3-5 qualitatively show the segmentation achieved using GC, DPRF and DPRF+GC.

Figure 5.6: **Failure cases**- Column 1 shows the articulated model overlaid on the images. Column-2 shows the pruned model that has rejected some part detections using DPRF. Columns 3-5 qualitatively show the segmentation achieved using GC, DPRF and DPRF+GC.

Chapter 6

Discussion

In this thesis, we have presented algorithms to automate several perception tasks that are crucial for situational awareness. In Chapter 2 we have presented a general framework to jointly classify objects that are observed from multiple vantage points in a distributed camera network. Our framework is well suited for wireless camera networks where each camera has limited processing capabilities. With the growing demand for security and surveillance, and lowering costs of connected devices, such smart camera sensors are starting to replace traditional CCTV based surveillance systems. In order to enable their large scale adoption, some crucial bottlenecks need to be addressed. One of the main constraints is the battery life - while it is possible to increase battery life of these sensors by using better quality batteries, another factor that makes significant difference is the choice of algorithms and processing on board the sensor. A second (and arguably more important) factor is the amount of data that needs to be wirelessly transmitted to a central station. In order to address the first constraint, we have evaluated several state-of-the-art computationally feasible feature detectors, and have chosen a representation that implicitly compresses these visual features to enable fast transmission. Further, we have drawn from recent developments in compressive sensing theory to formulate a distributed compression scheme that further minimizes the amount of data transmitted without loss of algorithmic performance. The degree of compression within our scheme can be dynamically selected, thereby making it more generalized. We have validated the performance of our system on very challenging data and under varying levels of compression.

One of the most challenging problems in computer vision, is that of segmenting objects of interest from the background in images. Successful separation of object from background can significantly help object detection and categorization. One benefit stems from being able to extract feature descriptors from segment regions, thereby improving the models of representation. In Chapter 3 we focus on this problem, and present a statistical method to segment informative features lying on foreground objects. Our method is scalable to large training image collections due to several novel algorithmic improvements that we have presented. The improved object models that we learn improves the recognition accuracy on our multi-view object database, while significantly reducing the model size by suppressing uninformative data in the model.

In Chapter 4 we have extended our static object recognition framework to the dynamic setting.

We present a joint human action detection and categorization framework for multi-view wireless smart camera networks. Human activity detection and categorization is an extremely challenging problem. This is because humans can exhibit multiple pose configurations, can be present in multiple scales with regards to the image sensor, and can be occluded by their own body parts or by scene objects. The ability to successfully detect humans, track them over multiple frames, and simultaneously recognize certain actions of interest automatically can significantly help automate several situational awareness tasks ranging from security and surveillance, to safety and transportation. We have presented a framework for distributed human action detection and categorization that builds upon our object recognition pipeline. In our approach, the computational overhead of extending inference to the temporal setting is localized to the central base station and does not increase number of operations on board the camera sensors. This stability in the types and number of operations on board the smart camera sensors enables easy deployment of such wireless smart sensors. We have successfully tested the efficacy of our joint activity detection and categorization framework on multiple challenging datasets.

Finally, in Chapter 5 we present a joint framework for general deformable object detection, segmentation and categorization. The types of models we consider are general and encompass rigid and non rigid objects. Our fusion of the top-down task of object detection and bottom-up task of image segmentation provides a very holistic and principled approach for addressing several challenging perception problems that arise in situational awareness applications. We experimentally validate our approach on a very challenging image dataset of humans in various poses with very challenging backgrounds. We strongly believe that the next wave of algorithms for automated situation awareness will focus of combining top-down and bottom-up information.

6.1 Future Work

Our investigations have opened up several avenues for future research. To begin, we strongly believe that the quality of object and landmark detectors can significantly improve with better models of representation. While simple template occurrence based models such as the Bag-of-words framework work reasonably well, they are mainly appealing in our distributed recognition setting as they are computationally very efficient. However, models that consider the geometry of the underlying object by accounting for the co-occurrence of part templates and the underlying 3D shape have been proven to be more precise in literature. Being able to extract such granular and informative representations from images can significantly improve the object detection and categorization performance.

We have discussed the importance of detecting humans precisely for several situation awareness tasks, and presented graphical model based frameworks for improving human detection in static images by fusing detection and segmentation. Although our method works well for static images, it does not scale to the dynamic setting where image pixels corresponding to a human in multiple image frames are all related to each other via the underlying kinematics of the human performing the action. Although precise modeling of human action kinematics has received some attention in the bio-mechanics communities, there is still significant room for research as there are several open

problems. One such problem is the ability to model any human action, and improve the parameters of this model using low-level bottom up image cues. Such joint frameworks for kinematic model fitting, action detection and segmentation can significantly improve the precision of automated perception systems for situation awareness.

Finally, Information extracted from the environment must be presented in a concise and comprehendible manner. In several complex and dynamic scenarios where situation awareness in critical for guaranteeing safety of humans, it is important to present information to decision-makers in a manner that is easy to comprehend. When data is transmitted from several sources at different locations and at different instances of time, a large amount of information is generated. This deluge of information can significantly affect the timely responses of decision-makers if it is not easy to visualize. Developing a seamless visualization framework can help bridge the gap between machine perception and human comprehension, thereby leading to truly autonomous situation awareness with humans in-the-loop.

Bibliography

[1] A. Beck and M. Teboulle. “A fast iterative shrinkage-thresholding algorithm for linear inverse problems”. In: *SIAM Journal on Imaging Sciences* 2.1 (2009), pp. 183–202.

[2] A. Thomas et al. “Towards multi-view object class detection”. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. 2006.

[3] A. Yang et al. *Fast ℓ_1-minimization algorithms and an application in robust face recognition: a review*. Tech. rep. UCB/EECS-2010-13. UC Berkeley, 2010.

[4] A. Yang et al. “Multiple-view object recognition in band-limited distributed camera networks”. In: *Proceedings of the ACM/IEEE International Conference on Distributed Smart Cameras*. 2009.

[5] Shivani Agarwal, Aatif Awan, and Dan Roth. “Learning to detect objects in images via a sparse, part-based representation”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.11 (2004), pp. 1475–1490.

[6] Saad Ali and Mubarak Shah. “Human action recognition in videos using kinematic features and multiple instance learning”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32.2 (2010), pp. 288–303.

[7] Pablo Arbeláez et al. “Semantic segmentation using regions and parts”. In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE. 2012, pp. 3378–3385.

[8] Dimitri P Bertsekas. “Nonlinear programming”. In: (1999).

[9] Moshe Blank et al. “Actions as space-time shapes”. In: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. Vol. 2. IEEE. 2005, pp. 1395–1402.

[10] Lubomir Bourdev and Jitendra Malik. “Poselets: Body part detectors trained using 3d human pose annotations”. In: *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE. 2009, pp. 1365–1372.

[11] Thomas Brox et al. “Object segmentation by alignment of poselet activations to image contours”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 2225–2232.

[12] C. Christoudias, R. Urtasun, and T. Darrell. "Unsupervised feature selection via distributed coding for multi-view object recognition". In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. 2008.

[13] C. Yeo, P. Ahammad, and K. Ramchandran. "Rate-efficient visual correspondences using random projections". In: *Proceedings of the IEEE International Conference on Image Processing*. 2008.

[14] E. Candès and T. Tao. "Near optimal signal recovery from random projections: Universal encoding strategies?" In: *IEEE Transactions on Information Theory* 52.12 (2006), pp. 5406–5425.

[15] Joao Carreira and Cristian Sminchisescu. "Cpmc: Automatic object segmentation using constrained parametric min-cuts". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 34.7 (2012), pp. 1312–1328.

[16] David Crandall, Pedro Felzenszwalb, and Daniel Huttenlocher. "Spatial priors for part-based recognition using statistical models". In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1. IEEE. 2005, pp. 10–17.

[17] Gabriella Csurka et al. "Visual categorization with bags of keypoints". In: *Workshop on statistical learning in computer vision, ECCV*. Vol. 1. 2004, p. 22.

[18] M. Wakin D. Baron et al. "Distributed compressed sensing". In: *preprint* (2005).

[19] D. Donoho. "For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution". In: *Communications on Pure and Applied Math* 59.6 (2006), pp. 797–829.

[20] D. Nistér and H. Stewénius. "Scalable recognition with a vocabulary tree". In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. 2006.

[21] D. Slepian and J. Wolf. "Noiseless coding of correlated information sources". In: *IEEE Transactions on Information Theory* 19 (1973), pp. 471–480.

[22] Navneet Dalal and Bill Triggs. "Histograms of oriented gradients for human detection". In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1. IEEE. 2005, pp. 886–893.

[23] Alexandre d'Aspremont et al. "A direct formulation for sparse PCA using semidefinite programming". In: *SIAM review* 49.3 (2007), pp. 434–448.

[24] Thomas Deselaers, Daniel Keysers, and Hermann Ney. "Discriminative training for object recognition using image patches". In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 2. IEEE. 2005, pp. 157–162.

[25] Laurent El Ghaoui, Vivian Viallon, and Tarek Rabbani. "Safe feature elimination in sparse supervised learning". In: *CoRR* (2010).

[26] Mica R Endsley. "Toward a theory of situation awareness in dynamic systems". In: *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37.1 (1995), pp. 32–64.

[27] Boris Epshtein and S Uliman. "Feature hierarchies for object classification". In: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. Vol. 1. IEEE. 2005, pp. 220–227.

[28] Alireza Fathi and Greg Mori. "Action recognition by learning mid-level motion features". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–8.

[29] Li Fei-Fei and Pietro Perona. "A bayesian hierarchical model for learning natural scene categories". In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 2. IEEE. 2005, pp. 524–531.

[30] Pedro F Felzenszwalb and Daniel P Huttenlocher. "Pictorial structures for object recognition". In: *International Journal of Computer Vision* 61.1 (2005), pp. 55–79.

[31] Pedro F Felzenszwalb and David McAllester. "Object detection grammars." In: *ICCV Workshops*. 2011, p. 691.

[32] Pedro F Felzenszwalb et al. "Object detection with discriminatively trained part-based models". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32.9 (2010), pp. 1627–1645.

[33] Pedro F Felzenszwalb et al. "Object detection with discriminatively trained part-based models". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32.9 (2010), pp. 1627–1645.

[34] Robert Fergus, Pietro Perona, and Andrew Zisserman. "A visual category filter for google images". In: *Computer Vision-ECCV 2004*. Springer, 2004, pp. 242–256.

[35] Robert Fergus, Pietro Perona, and Andrew Zisserman. "Object class recognition by unsupervised scale-invariant learning". In: *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2003, pp. II–264.

[36] Carolina Galleguillos, Andrew Rabinovich, and Serge Belongie. "Object categorization using co-occurrence, location and appearance". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–8.

[37] Stephen Gould, Tianshi Gao, and Daphne Koller. "Region-based segmentation and object detection". In: *Advances in neural information processing systems*. 2009, pp. 655–663.

[38] Stephen Gould et al. "Multi-class segmentation with relative location prior". In: *International Journal of Computer Vision* 80.3 (2008), pp. 300–316.

[39] H. Bay et al. "SURF: Speeded Up Robust Features". In: *Computer Vision and Image Understanding* 110.3 (2008), pp. 346–359.

[40] Minh Hoai, Zhen-Zhong Lan, and Fernando De la Torre. "Joint segmentation and classification of human actions in video". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 3265–3272.

[41] J. Lee. *Libpmk: A pyramid match toolkit*. Tech. rep. MIT-CSAIL-TR-2008-017. MIT, 2008.

[42] J. Philbin and A. Zisserman. *The Oxford buildings dataset*. http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/.

[43] J. Philbin and J. Sivicand A. Zisserman. "Geometric Latent Dirichlet Allocation on a Matching Graph for Large-scale Image Datasets". In: *International journal of computer vision* (2010).

[44] A. Zisserman J. Sivic. "Video Google: A text retrieval approach to object matching in videos". In: (2003).

[45] J. Yang and Y. Zhang. "Alternating direction algorithms for ℓ_1-problems in compressive sensing". In: *(preprint) arXiv:0912.1185* (2009).

[46] Yu-Gang Jiang, Chong-Wah Ngo, and Jun Yang. "Towards optimal bag-of-features for object categorization and semantic video retrieval". In: *Proceedings of the 6th ACM international conference on Image and video retrieval*. ACM. 2007, pp. 494–501.

[47] Ian T Jolliffe, Nickolay T Trendafilov, and Mudassir Uddin. "A modified principal component technique based on the LASSO". In: *Journal of Computational and Graphical Statistics* 12.3 (2003), pp. 531–547.

[48] Frédéric Jurie and Cordelia Schmid. "Scale-invariant shape features for recognition of object categories". In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2004, pp. II–90.

[49] Jan Knopp, Josef Sivic, and Tomas Pajdla. "Avoiding confusing features in place recognition". In: *Computer Vision–ECCV 2010*. Springer, 2010, pp. 748–761.

[50] Vladimir Kolmogorov and Ramin Zabin. "What energy functions can be minimized via graph cuts?" In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.2 (2004), pp. 147–159.

[51] M Pawan Kumar, Philip HS Torr, and Andrew Zisserman. "An object category specific MRF for segmentation". In: *Toward Category-Level Object Recognition*. Springer, 2006, pp. 596–616.

[52] L. Fei-Fei, R. Fergus, and P. Perona. "Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories". In: *IEEE CVPR Workshop on Generative-Model based Vision*. 2004.

[53] Lubor Ladicky et al. "Associative hierarchical crfs for object class image segmentation". In: *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE. 2009, pp. 739–746.

[54] Lubor Ladicky et al. "Graph cut based inference with co-occurrence statistics". In: *Computer Vision–ECCV 2010*. Springer, 2010, pp. 239–253.

[55] L'ubor Ladickỳ et al. "What, where and how many? combining object detectors and crfs". In: *Computer Vision–ECCV 2010*. Springer, 2010, pp. 424–437.

[56] Tian Lan, Yang Wang, and Greg Mori. "Discriminative figure-centric models for joint action localization and recognition". In: *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE. 2011, pp. 2003–2010.

[57] Ivan Laptev et al. "Learning realistic human actions from movies". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–8.

[58] Diane Larlus and Frédéric Jurie. "Combining appearance models and markov random fields for category level object segmentation". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–7.

[59] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories". In: *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2006, pp. 2169–2178.

[60] Bastian Leibe and Bernt Schiele. "Scale-invariant object categorization using a scale-adaptive mean-shift search". In: *Pattern Recognition*. Springer, 2004, pp. 145–153.

[61] Jingen Liu, Jiebo Luo, and Mubarak Shah. "Recognizing realistic actions from videos in the wild". In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE. 2009, pp. 1996–2003.

[62] D. Lowe. "Object recognition from local scale-invariant features". In: *Proceedings of the IEEE International Conference on Computer Vision*. 1999.

[63] Yi Ma et al. "An Invitation to 3-D Vision: From Images to Geometric Models". In: (2010).

[64] Lester W Mackey. "Deflation methods for sparse PCA". In: *Advances in neural information processing systems*. 2008, pp. 1017–1024.

[65] Nikhil Naikal, Dheeraj Singaraju, and S Shankar Sastry. "Using models of objects with deformable parts for joint categorization and segmentation of objects". In: *Computer Vision–ACCV 2012*. Springer Berlin Heidelberg, 2013, pp. 79–93.

[66] Nikhil Naikal, Allen Y Yang, and S Shankar Sastry. "Informative feature selection for object recognition via sparse PCA". In: *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE. 2011, pp. 818–825.

[67] Nikhil Naikal, Allen Y Yang, and S Shankar Sastry. "Towards an efficient distributed object recognition system in wireless smart camera networks". In: *Information Fusion (FUSION), 2010 13th Conference on*. IEEE. 2010, pp. 1–8.

[68] Yurii Nesterov. "A method of solving a convex programming problem with convergence rate O (1/k2)". In: *Soviet Mathematics Doklady*. Vol. 27. 2. 1983, pp. 372–376.

[69] Juan Carlos Niebles, Chih-Wei Chen, and Li Fei-Fei. "Modeling temporal structure of decomposable motion segments for activity classification". In: *Computer Vision–ECCV 2010*. Springer, 2010, pp. 392–405.

[70] Juan Carlos Niebles, Hongcheng Wang, and Li Fei-Fei. "Unsupervised learning of human action categories using spatial-temporal words". In: *International Journal of Computer Vision* 79.3 (2008), pp. 299–318.

[71] Björn Ommer and Joachim M Buhmann. "Learning compositional categorization models". In: *Computer Vision–ECCV 2006*. Springer, 2006, pp. 316–329.

[72] Andreas Opelt, Axel Pinz, and Andrew Zisserman. "Learning an alphabet of shape and appearance for multi-class object detection". In: *International Journal of Computer Vision* 80.1 (2008), pp. 16–44.

[73] P. Chen et al. "CITRIC: A low-bandwidth wireless camera network platform". In: *Proceedings of the International Conference on Distributed Smart Cameras*. 2008.

[74] P. Turcot and D. Lowe. "Better matching with fewer features: The selection of useful features in large database recognition problems". In: *ICCV Workshop on Emergent Issues in Large Amounts of Visual Data*. 2009.

[75] Caroline Pantofaru, Cordelia Schmid, and Martial Hebert. "Object recognition by integrating multiple image segmentations". In: *Computer Vision–ECCV 2008*. Springer, 2008, pp. 481–494.

[76] M Pawan Kumar, Philip Torr, and Andrew Zisserman. "Extending Pictorial Structures for Object Recognition". In: (2004).

[77] Andrew Rabinovich et al. "Objects in context". In: *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE. 2007, pp. 1–8.

[78] Deva Ramanan. "Learning to parse images of articulated bodies". In: *Advances in neural information processing systems*. 2006, pp. 1129–1136.

[79] M.D. Rodriguez, J. Ahmed, and M. Shah. "Action MACH a spatio-temporal Maximum Average Correlation Height filter for action recognition". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008.

[80] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "Grabcut: Interactive foreground extraction using iterated graph cuts". In: *ACM Transactions on Graphics (TOG)*. Vol. 23. 3. ACM. 2004, pp. 309–314.

[81] S. Nene, S. Nayar, and H. Murase. *Columbia object image library (COIL-100)*. Tech. rep. Columbia University CUCS-006-96, 1996.

[82] S. Wright, R. Nowak, and M. Figueiredo. "Sparse reconstruction by separable approximation". In: *IEEE International Conference on Acoustics, Speech and Signal Processing*. 2008.

[83] Scott Satkin and Martial Hebert. "Modeling the temporal extent of actions". In: *Computer Vision–ECCV 2010*. Springer, 2010, pp. 536–548.

[84] Christian Schuldt, Ivan Laptev, and Barbara Caputo. "Recognizing human actions: a local SVM approach". In: *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. Vol. 3. IEEE. 2004, pp. 32–36.

[85] Qinfeng Shi et al. “Discriminative human action segmentation and recognition using semi-markov model”. In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, pp. 1–8.

[86] Jamie Shotton et al. “Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context”. In: *International Journal of Computer Vision* 81.1 (2009), pp. 2–23.

[87] Dheeraj Singaraju and René Vidal. “Using global bag of features models in random fields for joint categorization and segmentation of objects”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 2313–2319.

[88] Josef Sivic et al. “Discovering objects and their location in images”. In: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. Vol. 1. IEEE. 2005, pp. 370–377.

[89] Cristian Sminchisescu, Atul Kanaujia, and Dimitris Metaxas. “Conditional models for contextual human motion recognition”. In: *Computer Vision and Image Understanding* 104.2 (2006), pp. 210–220.

[90] Noah Snavely, Steven M Seitz, and Richard Szeliski. “Modeling the world from internet photo collections”. In: *International Journal of Computer Vision* 80.2 (2008), pp. 189–210.

[91] Lingling Tao et al. “Sparse hidden markov models for surgical gesture classification and skill evaluation”. In: *Information Processing in Computer-Assisted Interventions*. Springer, 2012, pp. 167–177.

[92] Yicong Tian, Rahul Sukthankar, and Mubarak Shah. “Spatiotemporal Deformable Part Models for Action Detection”. In: ().

[93] Antonio Torralba, Kevin P Murphy, and William T Freeman. “Contextual models for object detection using boosted random fields”. In: *Advances in neural information processing systems*. 2004, pp. 1401–1408.

[94] Ioannis Tsochantaridis et al. “Large margin methods for structured and interdependent output variables”. In: *Journal of Machine Learning Research*. 2005, pp. 1453–1484.

[95] Ioannis Tsochantaridis et al. “Large margin methods for structured and interdependent output variables”. In: *Journal of Machine Learning Research*. 2005, pp. 1453–1484.

[96] Pavan Turaga et al. “Machine recognition of human activities: A survey”. In: *Circuits and Systems for Video Technology, IEEE Transactions on* 18.11 (2008), pp. 1473–1488.

[97] V. Chandrasekhar et al. “CHoG: Compressed histogram of gradients A low bit-rate feature descriptor”. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. 2009.

[98] V. Ferrari, T. Tuytelaars, and L. Van Gool. “Integrating multiple model views for object recognition”. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. 2004.

[99] Jakob Verbeek and William Triggs. "Scene segmentation with crfs learned from partially labeled images". In: (2007).

[100] Michel Vidal-Naquet and Shimon Ullman. "Object Recognition with Informative Features and Linear Classification." In: *ICCV*. Vol. 3. 2003, p. 281.

[101] Heng Wang et al. "Action recognition by dense trajectories". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 3169–3176.

[102] Ying Wang, Kaiqi Huang, and Tieniu Tan. "Multi-view gymnastic activity recognition with fused hmm". In: *Computer Vision–ACCV 2007*. Springer, 2007, pp. 667–677.

[103] Daniel Weinland, Edmond Boyer, and Remi Ronfard. "Action recognition from arbitrary views using 3d exemplars". In: *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE. 2007, pp. 1–7.

[104] Thomas Wiegand et al. "Overview of the H. 264/AVC video coding standard". In: *Circuits and Systems for Video Technology, IEEE Transactions on* 13.7 (2003), pp. 560–576.

[105] Alan S Willsky et al. "Nonparametric Bayesian learning of switching linear dynamical systems". In: *Advances in Neural Information Processing Systems*. 2008, pp. 457–464.

[106] John Winn and Jamie Shotton. "The layout consistent random field for recognizing and segmenting partially occluded objects". In: *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. Vol. 1. IEEE. 2006, pp. 37–44.

[107] Chen Wu, Amir Hossein Khalili, and Hamid Aghajan. "Multiview activity recognition in smart homes with spatio-temporal features". In: *Proceedings of the Fourth ACM/IEEE International Conference on Distributed Smart Cameras*. ACM. 2010, pp. 142–149.

[108] Y. Zhang, A. d'Aspremont, and L. El Ghaoui. "Sparse PCA: convex relaxations, algorithms and applications". In: *(preprint) arXiv:1011.3781* (2010).

[109] Yi Yang and Deva Ramanan. "Articulated pose estimation with flexible mixtures-of-parts". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 1385–1392.

[110] Yi Yang and Deva Ramanan. "Articulated pose estimation with flexible mixtures-of-parts". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 1385–1392.

[111] Yi Yang et al. "Layered object detection for multi-class segmentation". In: *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*. IEEE. 2010, pp. 3113–3120.

[112] Z. Cheng, D. Devarajan, and R. Radke. "Determining vision graphs for distributed camera networks using feature digests". In: *EURASIP Journal on ADvances in Signal Processing* (2007), pp. 1–11.

[113] Z. Lin et al. "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices". In: *UIUC Tech Report: UILU-ENG-09-2215* (2009).

[114] Jianguo Zhang et al. “Local features and kernels for classification of texture and object categories: A comprehensive study”. In: *International journal of computer vision* 73.2 (2007), pp. 213–238.

[115] Zhenyue Zhang, Hongyuan Zha, and Horst Simon. “Low-rank approximations with sparse factors I: Basic algorithms and error analysis”. In: *SIAM Journal on Matrix Analysis and Applications* 23.3 (2002), pp. 706–727.

[116] Zoran Zivkovic. “Improved adaptive Gaussian mixture model for background subtraction”. In: *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. Vol. 2. IEEE. 2004, pp. 28–31.

[117] Hui Zou, Trevor Hastie, and Robert Tibshirani. “Sparse principal component analysis”. In: *Journal of computational and graphical statistics* 15.2 (2006), pp. 265–286.