EVENT CLASSIFICATION IN A KITCHEN DOMAIN

JAWAD TAYYUB

MSC ADVANCED COMPUTER SCIENCE

Session 2012/2013

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of student)___________________________

Summary

Event analysis and recognition is a major in computer vision today. Lots of research has undergone in this area to produce robust and accurate methods for recognition of activities in a natural way. Much of previous work has focused on the use of arbitrary set of action sequences such as ’jumping’ or ’running’ for recognition. In this project we aim to extend on that and focus of recognition of real life every day tasks represented in a hierarchical structure.

Most complex natural activities are composed of a set of simple actions which in turn are composed of simpler actions and so on. Therefore, an activity can be naturally represented in a hierarchical structure. Furthermore, the temporal order of actions could also be represented using the same hierarchy. We propose a new event recognition model where the activity is represented in such a hierarchy. In order to recognize the activity, the most basic lowest level events need to be recognized only. The recognition of higher level events would depend on the specific order of low level events to be recognized as specified by the model structure. We could therefore recognize an entire activity through inference.

Lastly, a machine learned system is proposed that would automatically learn the event model of a complex activity given ground truth data. The project aims to evaluate these systems using an an every day real life activity data set which comprises of events commonly taking place in a Kitchen Domain.

Acknowledgements

Most of all I'd like to thank my project supervisor Professor Anthony Cohn for the knowledge, encouragement, freedom, invaluable advice and patient guidance he has provided me through out the course of this project. It has been an honour to have worked under the supervision of a person who cared so much about my work and was readily available for advice and comments even in the most difficult situations.

I would like to extend gratitude towards Dr. Muralikrishna Sridhar and Dr. Aryana Tavanai for their tireless efforts and uninterrupted support throughout the development of the project. Both researchers have been extremely friendly and helpful and never shy to answer a mundane question. The knowledge I have gained from them during the development of this project is priceless.

I would also like to thank my father for his uncanny support, my mother for her undying love and lastly my beloved sister for her support throughout my academic life.

Finally I would thank my friends here at University of Leeds for the support and encouragement provided in time of hardship and peace.

Contents

1 Introduction 1

1.1 Overview 1

1.2 Aims & Objectives 2

1.3 Minimum Requirements and Deliverables 3

1.4 Project Management 3

1.5 Research Methodology 3

1.6 Framework Overview 4

1.7 Research Questions 5

1.8 Novelty 5

1.9 Thesis Overview 6

2 Literature Review 7

2.1 Introduction 7

2.2 Terminology 8

2.3 Techniques 8

2.3.1 Template-based approach 9

2.3.2 Optical-flow approach 10

2.3.2.1 Space-Time Interest Point 10

2.3.3 State-Space approach 11

2.3.3.1 Hidden Markov Model 11

2.3.3.2 Bayesian Networks 12

2.3.4 Modern Approaches 13

2.3.4.1 Activity recognition from movies 13

2.3.4.2 Egocentric Activity Monitoring 14

2.3.4.3 Unsupervised Activity Analysis 15

2.4 Relation to our work 17

2.5 Conclusion 18

3 Logical Hierarchical Event Modelling 19
3.1 Introduction . . . 19
3.2 Formulation . . . 19
3.3 Event Detection . . . 23
3.3.1 Multiple Hypothesis Event Detection . . . 23
3.4 Learning the Event Model . . . 24
3.5 Representation of Interactions . . . 25

4 Experiments 27
4.1 Introduction . . . 27
4.2 Dataset . . . 27
4.2.1 Introduction . . . 27
4.2.2 Motivation . . . 27
4.2.3 Data . . . 28
4.3 Implemented system . . . 30
4.3.1 Introduction . . . 30
4.3.2 Specific Framework . . . 30
4.3.3 Rules and Constraints . . . 32
4.3.3.1 Rules . . . 32
4.3.3.2 Constraints . . . 32
4.3.4 Conclusion . . . 33
4.4 Evaluation Method . . . 33
4.5 Exp 1: Performance of the Logical Hierarchical System . . . 34
4.5.1 Introduction . . . 34
4.5.2 Experiment . . . 34
4.5.3 Result . . . 34
4.5.4 Conclusion . . . 35
4.6 Exp 2: Logical Hierarchical System with added window size . . . 36
4.6.1 Experiment . . . 36
4.6.2 Results . . . 36
4.6.3 Conclusion . . . 36
4.7 Exp 3: Performance of the Logical Hierarchical System with Object inclusion . . . 37
4.7.1 Experiment . . . 37
4.7.2 Results . . . 38
4.7.3 Conclusion . . . 38
4.8 Exp 4: Performance of Logical Hierarchical system with Noise . . . 39
4.8.1 Experiment . . . 39
4.8.2 Results . . . 39
4.8.3 Conclusion . . . 40

4.9 Exp 4: Performance of multiple hypothesis system with Noise 40
4.9.1 Experiment 40
4.9.2 Results 41
4.9.3 Conclusion 41
4.10 Exp 4: Performance of the Machine-Learned event Recognition 42
4.10.1 Experiment 42
4.10.2 Results 42
4.10.3 Conclusion 42
4.11 Exp 4: Performance of the Machine-Learned event Recognition with Object Information 43
4.11.1 Experiment 43
4.11.2 Result 43
4.11.3 Conclusion 43

5 Conclusion and Further Work 45
5.1 Findings 45
5.2 Project Evaluation 46
5.2.1 Aims and Minimum Requirements 46
5.2.2 Research Questions 47
5.2.3 Exceeding Requirements 48
5.2.4 Conclusion 48
5.3 Further Work 48

Bibliography 50

A Personal Reflection 52

B Software Used 54

C Project Schedule 55

Chapter 1

Introduction

1.1 Overview

Video activity analysis is an active area of research in the computer vision field. Various recognition techniques are being developed to allow computer system to automatically recognize interesting events from digital videos. Modelling and recognizing human activity is a particularly interesting area of the broader activity analysis domain. Such recognition systems are proving to be revolutionary in the fields of security surveillance, robotics, monitoring systems, human computer interactions etc.

Humans perceive and understand an activity as a set of events that it is made of. For example, the activity of 'making coffee' is the process of getting a cup, getting coffee beans, grinding them up, mixing, brewing etc. This way of breaking down an activity into smaller simple actions could be used to effectively model a complex activity. The activity can be represented in a hierarchy of sub-actions that make up the activity. These sub-actions can be further broken down in simpler sub-actions. The process could repeat until we reach atomic actions of the activity. Thus a hierarchical representation is natural for representing many human activities.

Another aspect of human activities is that they arise when actions are applied to objects. For example, a actions such as *wash, peel, chop* could be to an object such as a carrot to produce events such as *wash carrot, chop carrot, peel carrot*. These actions could then be applied to another object - *wash onion, peel onion, chop carrot* etc. Thus actions may be naturally represented by logical predicates so that the objects constitute arguments for these predicates that correspond to actions.

In this project we explore the possibility of using such hierarchical representation to propose an event recognition model based on a logical structure. We then aim to build a pre-defined and machine learned event recognition system respectively, using the proposed event model. We demonstrate that such a model has a promising potential for recognizing complex activities using a table top setting dataset.

Another key feature of our system is that we infer actions by modelling interactions between an agent and the object in terms of qualitative spatial relationships. In particular we adopt sridhar's [15] set of spatial relationships for representing interactions, as it has shown promising results on more than one video dataset. We extend this approach to include both a hierarchical structure and a logical structure to events.

The final and most significant aspect of our work stems from the fact that many day today activities tend to have a fixed high level structure i.e. they tend to follow a script, like a cook follows a recipe, or waiter laying a table. Thus the cost of requiring large amounts of training and complex inference schemes using models such as Hierarchical Hidden Markov Models may overweight the minimal benefits that may account from modelling variability at mid and higher levels. Thus we have assumed a less flexible structure at these levels i.e. we assume that we have been given a script, in particular laying out a table by placing a tray, then cups and so on[1].

However, the problem of interpreting observations given a script still remains, as perceptual data tends to produce ambiguity. We propose a *logical hierarchical event model* for using such a script together with a *multiple hypothesis system* in order to address such ambiguities. The logical hierarchical event model encodes constraints with regard to action transitions and the passing of objects from one action to another. These constraints are used to resolve ambiguities using the multiple hypothesis system.

1.2 Aims & Objectives

The aim of this project is to develop a novel framework for event recognition. We aim to propose a real-time event recognition model. The model would be capable of passing arguments between events. It would also allow for multiple hypothesis to be maintained. These define the many possible events occurring at any time. We also aim to explore the system on a new and challenging dataset [17]. The dataset resembles a real world video with real world everyday actions. These are different from the clich datasets that resemble abstracted or artificial motion such as walking, running, jumping etc. This would allow pushing state-of-the-art activity analysis system to their limit.

Following is a formal list of our objectives:

- Develop a hierarchical logical event model for hierarchical detection and labelling of activities along with its sub-events.
- Implement a manually defined system from the proposed model. The system would use an instance of the model that would be specific for the kitchen dataset.

[1] Future work could focus on how to obtain which script to use given cues from the video.

- Develop a machine learned system that could automatically learn the event model given ground truth data instead for manually defining the model.
- The recognition system should label videos in real-time.
- Develop a system that allows for multiple hypothesis on detected events to be maintained in parallel and resolved upon receiving sufficient information.
- Further annotate TUM kitchen dataset to include object information as well as higher level event labels.

1.3 Minimum Requirements and Deliverables

The minimum requirements of the project are given as below:

1. Identification and further annotation of the TUM kitchen dataset videos to provide ground truth for experiments.
2. Creation of a rule-based system capable of recognising simple events such as picking up something or putting down something.
3. Creation of a system able to recognise complex events composed of simple events.

1.4 Project Management

A project plan was created at the beginning. At that time, there were rough details available on the exact nature of the project. The original project plan is presented in Appendix C Figure C.1. Also in Appendix C Figure C.2 presents the revised plan. In the revised plan, the length of 'Annotation of Videos' was extended substantially. This was due to improper documentation available for the video dataset. This required a long time to manually parse the video motion capture data to retrieve position of the human hands. This extension caused subsequent tasks in the project to be pushed further.

The earlier sections such as 'Literature Review', 'Introduction' etc. were managed well and within time constraints. The development of extensions section moved ahead after feedback from the assessor meeting. Overall the project was averagely managed.

1.5 Research Methodology

The research methodology adopted in this project is evolutionary prototyping. Using this approach, an initial prototype is built that employs very basic techniques and implementation. This first prototype is evaluated for faults and major issues. The prototype might then be modified iteratively where with each version, the evaluation is repeated and possible further features added until a system that satisfies the objectives is achieved. Since the objectives were somewhat uncertain in the beginning of the project, the prototypes development went through directional shifts during development. The evolutionary prototyping method suits to this modality of development where uncertainty exists in the specifics of the implementation details. With each implemented prototype, the results dictate the direction of the project.

The initial prototype was aimed at employing the simple single manually defined system. It worked by observing the change in distance measure between two objects in two consecutive frames to recognize basic events such as 'taking' or 'putting'. This prototype produced very noisy labelling of basic events primarily because of minor variation in object detection. This prototype was discarded as infeasable.

Following set of prototypes known as P_1 were developed with the employment of spatial relations as used by Sridhar [15] to model interactions. The prototype was then equipped with manually defined rules that guided the system to label basic events using the corresponding spatial relations. New prototypes were built with introduction of more advanced rules to achieve identification of higher level events. Finally, a multiple hypothesis component was introduced to the system that would allow for many event detections to be simultaneously kept as hypothesis as the video plays. These hypothesis would then be resolved using predefined resolution criteria whenever it is met. For example, at a certain frame two events such as 'taking x' and 'taking y' might be initialized as hypothesis of possible events happening. Only later in the video would we be able to label these events with certainty when the actual item taken is detected as being carried.

The next set of prototypes, known as P_2, would allow for the manually defined rules, that make up the model, to be learned using a machine learning technique given the ground truth. The motivation for this was to completely automate all aspects of the system and make it applicable to a variety of domains rather than just the one that was defined by the data set used. For example a new scene can be observed and given ground truth data with labelled events and activities, a set of rules that define the events can be obtained. These could then be used to recognize activities in new data. This machine-learned prototype was evaluated against the earlier manual prototypes to compare and contrast the performance of the two variations of the system.

1.6 Framework Overview

1. Data set acquisition: A challenging data set of videos with an activity was acquired. The dataset was chosen that comprises of activities that area happening in a real life scenarios.

2. Data Preparation: The data set available was pre processed to a form that would be compatible with the frameworks specifications. This included tasks such as 3D projection of coordinates, updating and further annotating of videos etc.

3. Qualitative Spatio-Temporal Representation: Interactions between objects are modelled in terms

of spatial features such as Discrete or Partial Overlap. Spatial representations are sufficient to identify interesting events in the scene.

4. Logical Hierarchical Event Model Development: A generative model is developed that would allow for activities to be recognized correctly.
5. Manual Event Recognition System. The model was used to program rules to define the events according to their corresponding spatial relationship representation. Rules are also written to define higher level events according to their corresponding lower level comprising events. The entire activity is defined as a hierarchy which can be thought of as a single instance of the event model.
6. Machine Learned Event Recognition System: The model and rules that define the activity in terms of events and sub-events were automatically learned instead of manually programmed. This was possible by using given ground truth data.

1.7 Research Questions

The following research question would be tackled:-

1. Could spatial relation be used to represent primitive interaction between objects ?
2. Could we incorporate and use action and object information together within a single recognition system.
3. Could we represent higher level events based on lower levels event and ultimately the entire activity as such a hierarchy ?
4. Can we use multiple hypothesis to resolve ambiguities arising from observation ?
5. Can the model be learned automatically given ground truth data ?
6. Would the automatic learned rules perform better than the manually programmed rules ?

1.8 Novelty

The novel aspect of our work is in interpret video activities in terms of a hierarchical and logical event model that uses a multiple hypothesis framework to resolve ambiguities in observation. Through literature review, we have seen that the models that are most similar to our proposed system is the logical model and the hierarchical hidden Markov model. These two models are well known for event classification using probabilistic analysis.

However, we put forth the hypothesis that the higher level structure for most activities are on the whole known in advance. For example, there is usually a standard way to lay a table. A more complex example would be, that once a type of dish is known in advance, the procedure at the higher levels largely gets fixed. A third example baking a cake or baking a pizza would have high level activities of 'baking' always the same. The changes would happen in the arguments. Hence, it may be overkill to model too much flexibility at the higher levels and pay the cost of extensive training data and complex training and inference schemes, that are characteristic of HMMs. Such HMMs become even more complex when they are extended with logic. While they are designed to handle variance in events, they do not naturally provide a way to encode constraints in the form of common sense rules that may be used to handle ambiguities in observation.

Our novel approach incorporates hierarchy and logic. While we do not focussing on modelling event variability, we focus on the problem of interpreting ambiguity in observations using a *multiple hypothesis approach*. The logical hierarchical event model encodes constraints with regard to action transitions and the passing of objects from one action to another, and are used to resolve ambiguities in observation.

We believe that our approach can be potentially developed into a top down bottom up framework, where observations about some objects and actions can be used to infer what kind of activity is going on i.e. to infer and fix the script, for example which dish is going to be prepared. This fixed script can then be used to interpret the video in terms of actions and sub-actions applied to different types of objects. We believe that this work is a starting point of such a promising approach towards activity analysis.

1.9 Thesis Overview

Chapter 2 presents the literature review. Various activity analysis techniques are researched, compared and contrasted. The chapter is written in a chronological order of the developments in the field of activity analysis and recognition.

Chapter 3 describes the framework of the proposed solution. A hierarchical logical event model is proposed in this chapter.

Chapter 4 presents the specific implementation of the framework described in chapter 3. It also consists of evaluation and experimentation of the framework. Conclusions about the proposed framework are then driven from experimentation results.

Chapter 5 gives the overall evaluation of the system. An analysis is done with the objective and requirements set out and the actual achievements. Research questions are answered and future work is proposed.

Chapter 2

Literature Review

2.1 Introduction

It has been a long fascination of scientists to teach computers to see the world like humans. This has resulted in massive amounts of research in the fields of computer vision, neuroscience and artificial intelligence. However, there are still several challenges to get past before a human-like artificial vision system can be developed. Video processing is a field that holds the key to realizing this fascination of a human-like vision system. In recent years, there has been a growing interest in this field of video processing. Video processing has been a natural step up from the vast research carried out on image processing. Tasks common in image processing, such as object recognition, image analysis, feature extraction etc., are now being extended to videos. Video analysis reaps similar benefits as image analysis. Video processing has a number of applications in todays world. Surveillance is probably one of the most significant fields to inspire video processing and recognition research. Other fields of such technology includes robotics, computer interaction, smart monitoring systems, prediction based surveillance and so on.

Image/Video processing boasts many challenging tasks. Some common challenges in the field of image/video processing are edge detection, object tracking, object categorization and/or classification, optical flow estimation etc. A common activity in image analysis is the recognition of objects. This is the process of identifying objects of interests in an image automatically. Object recognition is usually categorized as a high-level computer vision problem that makes use of many low-level vision techniques like edge detection, filtering etc. Successful work has been done to perform object recognition accurately and efficiently. Most noticeably, the SIFT descriptor developed by DG Lowe [8], has been a revolutionary technique to extract scale invariant features from images that can then be used to perform object recognition. This is an example of feature-based recognition. Belongie [2] developed a robust object recognition system using shape contexts. Correspondence or similarity measures are developed to identify and categorize new shapes. This is a pattern-based approach. There has been much development on the techniques presented for pattern-based or feature-based object recognition. When it comes to videos, the object recognition task on images becomes a directly corresponding activity recognition task on videos. As in images, the prominent interesting contents would be mostly objects, in videos they would be activities.

An accurate definition of activities was given by Lavee [7]. He describes an activity as those high-level semantic concepts that humans perceive as interesting when observing a video clip. In this literature review we will be describing a number of activity analysis techniques that have been developed in recent years.

2.2 Terminology

Before diving into the literature on the various activity analysis methods and techniques, let us first describe some terminology commonly used in the field.

Activity: An activity is described as a set of actions/events that are semantically meaningful to human beings. Activities can range from simple activities comprising of a few events such as making coffee to complex activities such as off loading luggage from an airplane comprising of many events.

Event: Activities are comprised of events. An event is a piece of simple interesting thing that happens in a scene for example reaching for an item.

2.3 Techniques

A vast amount of techniques have been developed to analyze videos and attempt to extract semantically meaningful knowledge from it. Various researchers have attempted to categorize the approaches in meaningful manner [7, 18]. Turaga [18] attempts to group them with respect to the type of recognition they perform. These are action/event based or activity based. Turaga [18] defined actions as simple primitive activities involving only one subject such as ’jumping’, ’jogging’ etc. He defines activities as a set of actions or events that involve multiple subjects of interest such as two humans shaking hands, throwing a ball etc. Clearly modeling and recognizing single actions is much simpler than recognizing complex activities, which are an aggregation of actions. In another approach, Lavee

[7] categorizes the various recognition techniques according to the actual technical models that the techniques employ. This is a much better description of the various techniques categorization since it clearly separates various approaches according to the actual mechanics and workings of each approach. For our purpose, we would attempt to categorize the activities in the chronological order of their invention.

Activity recognition techniques cover a great deal of literature. Turaga [18] summarizes the very general process of activity recognition in four steps:

1. Input a video, 2. Low-level feature extraction from the input video, 3. Some processing on low-level features to create simple descriptions of actions and 4. Recognition of the entire activity given these actions.

Machine Learning techniques have spearheaded the research in image and video analysis by a huge factor. Various machine learning approaches such as SVM classifiers, nearest neighbors, Bayes networks etc. have been a very complimentary addition to the traditional techniques of recognition. At the broadest level, the various activity recognition techniques can be split into supervised and unsupervised techniques. However, most of the techniques could be adapted to either category with slight modifications. Even though all techniques can be classified in one of these two categories individually, there will always be overlap for techniques that are essentially the same but adapt one of the two machine learning approaches. Therefore a different categorization would be required to identify the techniques separately. We are simply going to mention the approaches in a chronological order.

2.3.1 Template-based approach

Some of the earliest approaches in vision analysis and activity understanding are techniques that involve pattern-recognition and template matching. In simple terms, the action to be recognized is converted into low level shape representation called a template. New videos are compared against the set of pre-stored templates to find a match. An early work in template matching is done by Polana and Nelson [10]. They proposed a method of detecting and recognizing the human posture using a periodicity measure. First the human is tracked. Then a cropped sequence of videos where the human is present is constructed. Then the video is segmented into cycles which are combined to form an average cycle representing the repetitive motion. The motion is represented as a set or repeating body posture, for example walking has a periodic repetition. This is modeled by flow features extracted from the segments of the video which are averaged to produce a single feature. These become the templates for that action class. Recognition is simply matching the flow features obtained from new video to the templates in order to identify the action in the new video. Advancement on this approach is proposed by Bobick and Davis [3], who might have been one of the first to introduce the idea of temporal knowledge incorporation in the feature representation. They use a motion history image MHI that gives weights to the frames in the video. Newer frames get higher weights than older frames. The templates produced now take into account the temporal knowledge along with motion knowledge. This has benefits over previous approaches, since a more discriminating set of features is created.

These early template-based techniques however share some common criticism. They are good for a very well defined set of videos and arbitrary actions. A walking representation will be well defined if the video records the side of the person where the walking motion is clear and apparent. Otherwise there will be problematic representations. Another disadvantage is that these techniques are more sensitive to variance of the movement duration. Their simplicity and ease of implementation is one undeniable benefit.

2.3.2 Optical-flow approach

This method was invented early on but still is used quite frequently. It is quite often used on its own or in conjunction with other methods even today. Optical flow refers to the apparent motion of pixel in image frames in a video. Motion of pixels is a good way to judge the activity in a video. Optical flow is useful as it provides a description of the motion of object/subjects in a video and also the velocity at which they are moving. These can be combined to represent the motion in a descriptive manner. Drawbacks or challenges are constant illumination changes or noise in the video might cause false detections.

2.3.2.1 Space-Time Interest Point

This is an example of optical flow approach. We will discuss an activity recognition and video analysis method that is basically feature extraction from video. It is usually categorized as a pattern-recognition low-level feature extraction method. It closely relates to the famous SIFT feature detector [8] commonly used in image processing. The idea for this approach of video analysis was first proposed by Lapev [5]. Laptev suggested that the tradition approaches such as feature tracking were effective for a number of tasks but it had draw backs. They were good techniques for videos that had single consistent motion and unchanging background, lighting or other variations. The traditional approaches did not work too well with real world videos where the background is constantly changing and motion is not static but rather sudden and uneven. To provide a feature that is discriminative and captures maximum information from such real-world videos, the Space-Time Interest Point feature is proposed.

Its idea is simply taking the Harris Interest Point detector, well-known in 2-d image work, and altering it to account for three-dimensions where the third dimension is the temporal dimension. This feature detector works very similarly to the Harris Point detector. The Harris Point detector finds features in images where the pixel values are varying highly in a single frame of an image through gradient maximization. The space-time interest point STIP detector applies the same concept but also looks for highly varying image values on the temporal dimension i.e. from one frame to another. Intuitively, this approach should capture interest points of highly changing pixel values, such as corners or edges, at the same time as changing pixel values on a temporal scale, for example a leg moves swiftly. These changes are temporal changes and would be similarly detected.

The final feature is expressive enough to model the temporal motion as well as actual image interest points and discriminating at the same time. A model can then be learned by training the system on many different videos of human motion and clustering the extracted STIP features. A new video can then be recognized by matching its STIP features with the model to classify that action. However, since the nearest match is always considered as the classifying class of the new video using this approach, it does not account for a case where the video is not any of the modeled predefined classes.

Much work has been done using STIP features. New classification algorithms such as k-means are being combined with the STIP feature approach to produce robust models of activity representation. Schuldt [12] uses STIP features along with SVM classification to produce a very robust system for classifying six different human-actions.

2.3.3 State-Space approach

A large number of approaches developed can be categorized under the state-space approach category. State-space approaches define every element, such as posture or atomic action such as lifting or sitting, of a video as a state. These different states are connected to each other with some probability defining the likelihood of the state going to another state. An activity or series of actions are defined as a route between the different states. The joint probabilities of the motion can be calculated and maximized to classify an activity. State-space approaches differ from the ones describe before in terms of the complexity of videos. In previously described approaches, videos were simple actions like walking, running etc. With state-space approaches it is possible to model more complex actions like juggling a ball or preparing coffee etc.

Lavee [7] and Turaga [18] provide a very good description of the various state-space methods currently in use today. We will summarize those definitions and compare them next.

2.3.3.1 Hidden Markov Model

Hidden Markov Model HMM is one of the most popular state-space models. A state-space model is a probabilistic graphical model that defines dependence, in terms of probability, between the states and measurements. In HMM models, it is assumed there is finite number of states in the state space.

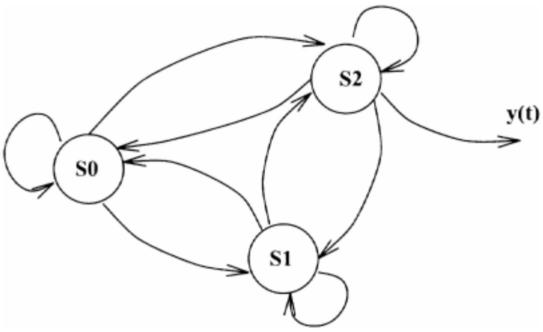


Figure 2.1: A basic Hidden Markov Model [Aggarwal et al 1998]

Figure 2.1 shows a basic structure of a Hidden Markov Model. Moving from one state to the next is modeled as probability distributions that judge the probability of the move. The states of the HMM are usually abstract. Parameters of the model can be learned by training on various numbers of videos or introducing manual background knowledge. Test data can be evaluated by determining the probability of the data being produced by each of the HMM models. The maximum probability model will be the classification of the data. Sridhar [16] makes use of an HMM to search for the most likely interpretation of the different tracks of motion that were extracted. More detailed discussion of Sridhars work is done in later parts of the report. Yamato [20] is probably the earliest work using HMMs to recognize the categories of tennis strokes being played. The approach was more robust in detection of activity in a noisy video than previous approaches. The results were quite impressive with recognition rates as high as 97 percent. Significant improvement in the result was found by increasing training dataset.

HMM models complex activities which inherently make the model complex. With increased number of states, the parameters of the model increase and it requires greater number of training examples to find the parameters. Traditionally HMM states only depend on the state before, but newer variations of HMM known as n-order HMM can have state dependencies from many previous states. This makes the model very complex and computationally expensive.

2.3.3.2 Bayesian Networks

Another state space approach is using Bayesian networks BNs. They are developed to deal with the issue of uncertainty in video events. There is always uncertainty as to which event will occur when. Probabilities are introduced in this domain to tackle with this challenge and work on most likely events. BNs can be defined as acyclic graphs where each represents a random variable. The structure of the graph allows for representing joint probabilities for all variables. Estimation of unknown variables can be done using values of known variables. BNs can have a more complex structure of conditional probabilities than the simple traditional HMMs. Wang [19] makes use of hierarchical Bayesian models to connect three different elements of the video. This is a new approach where more than one feature is being used to model activities. Low-level visual features, simple basic actions and interactions are modeled altogether. Atomic actions are modeled by clustering different moving pixels (low-level features). Similarly interactions are modeled by clustering many atomic actions together. This approach has been proven useful in detecting anomalies in video, recognizing activities in clutter scenes and motion segmentation.

Other state space approaches exist such as petri-nets or grammars. More recently, innovative data structures are used by Sridhar [16] called interaction graphs and by Hamid [4] called suffix trees. State-space approaches are useful for modeling a complex scene with multiple actors but they share a drawback. Since these approaches use probabilistic non-linear models, they do not always have a closed form solution. A global optimum is sought out but it may take a long time and computation power to find it. Local optima that are viable are often used. However various techniques such as simulated annealing exist to aid in the optima search.

2.3.4 Modern Approaches

New research in video analysis is focused more and more on analyzing and processing realistic motion for example setting up a table top [17] in a kitchen setting. This is in contrast with artificial periodic motion such as jumping, kicking etc. There is also a trend in video analysis research that focuses on real-time video processing rather than complete video sequence processing. There exists a lot of literature that tackles real-time video processing and activity analysis challenges [16, 11, 9]. This is crucial in video surveillance application since the base idea is to automatically detect and act upon an interesting event (e.g. an anomaly) as it happens or preferably before it happens [12].

2.3.4.1 Activity recognition from movies

An innovative idea was proposed by Laptev [6] as an attempt at recognition of realistic motion. The idea is to use text dialog that goes alongside the video to deduce the activities. Most commonly this is found in movies where the dialog of the characters is available alongside as subtitles. Since text describing the action in a movie scene can be quite variable, machine learning techniques are applied to classify the actions. A bag-of-feature approach is used where the scene description is represented as a high-dimensional vector. Other features such as space-time features and spatio-temporal bag-of-features are used to build a space-time grid to represent the scene are used. These detailed representations produce very good results for simple actions such as walking, jogging etc. For more realistic actions from movie scenes such as kissing, answering the phone etc., the results are decent but not the impressive. Their reasoning for this is incorrectly labeled training data.

2.3.4.2 Egocentric Activity Monitoring

Another interesting and innovative idea for activity recognition was put forward by Behrera [1]. The analysis of activity proposed by Behrera is in-sync with the new trend in activity analysis research mentioned above. Behreras work focuses on an egocentric situation that uses first-person wearable cameras. The cameras in this research are worn by a factory worker. They provide a first person video of the factory workers actions. Camera data is further complemented by inertia motion unit IMU data, which are worn on the wrists of the factory worker. The aim of this work is to provide on-the-spot instructions and advice to workers through a wearable HMD as they perform the task. The system is expected to recognize and analyze activities in real-time to be able to advice the factory workers. Behrera reports that the posture of a person alone and the way it changes may be enough to represent periodic artificial activities such as jogging or running as in previous work [12], but it is not enough to represent more fine-grain and realistic everyday activities such as making coffee, hammering a nail. A deeper representation is needed to accurately model these fine grain activities. The feature proposed in this work models the object to object relationship along with object to wrist relation. This spatiotemporal relationship between the workers hands and objects provides important cues for the activities being performed. The various aspects modeled are the speed at which the hand approaches the objects and interaction between the hands and objects. An important aspect of this approach is that these spatiotemporal relations are learned by the system rather than being manually defined. This allows for the system to automatically extract interesting features from the video feed.

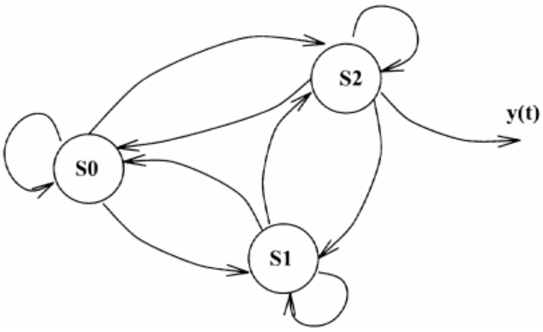


Figure 2.2: A hierarchical structure of Behrera's approach [Behrera et al 2012]

Figure 2.2 accurately describes the hierarchical structure of the method by which features are extracted for an activity from a window of a video. At layer 1 an activity y can be described as a set of temporal events e shown in layer 2. Events are described as a histogram of relations that define that event as shown in layer 3. The histogram counts the number of spatiotemporal relations for a particular event using a codebook that has been generated earlier by simple k-means clustering of all relations. This technique is known as bag-of-relations approach which is closely related to the well-known bag-of -features approach [6] or bag-of-words approach [13]. An interesting addition to the bag-of-relations approach is that the there is another histogram that captures the relations from the IMUs. The data from IMUs help model the relations between the parts of the human body when carrying out an activity. For example the captured relations are the elbow-wrist relation or the shoulder-torso relation etc. These motions are captures as angles, rates and acceleration relations between the various body parts. According to Behrera, using this combination of histograms creates a representation that would be able to represent each activity uniquely enough so that a new activity could be quickly recognized and categorized.

The results obtained from this approach are decent. Experimentation is performed on two tasks. Experimentation is done using simple vision histograms, simple IMU histograms, simple STIP [5] features and a combination of them. Broadly speaking, it has been reported that using the pair wise approach (any two together) generally produces better results than any one alone. This is simply because multiple features allow for greater and higher level of representation of activities. This improves the system to discriminate between activities when classifying them.

Some drawbacks of the approach include lack of temporal representation. The order in which events happen is not detailed stored in this representation. This causes confusion between discriminating activities that are temporally different but eventually similar. Research by Sridhar [16] makes use of activity graphs where the temporal relations are stored in great details at the top level. Details on this work will be provided later. Another weakness of this representation is its inability to discriminate between events that are similar in spatio-temporal relationships but different to a human eye. These include for example for picking up a nail and placing it on table or picking up screw and placing it on table. If the object is not well detected both these activities will be categorized as the same one. Lastly this system of detection and recognizing requires an intrusive setup where the user is required to wear equipment. This may not be suitable for natural surveillance system but perhaps in a factory setting with workers; it is acceptable in this case.

2.3.4.3 Unsupervised Activity Analysis

Unsupervised activity analysis involves automatically inferring an activity without prior knowledge. Much of research has been primarily focused on supervised techniques of activity analysis for the obvious reasons of simplicity. But more recently work has been done to explore the unsupervised methods as well [16, 19, 4].

Sridhar [16] work is particularly interesting as it makes use of intricate graph data structures to preserve the dynamics of an activity. Behrera [1] work described earlier can be thought of as a simplification of this work. In Sridhars approach, the problems being tackled are complex. The challenge is to perform video analysis on video obtained from a static camera pointing a busy scene with many subjects and objects and automatically learning interesting events from observation. The main assumption in this work is that many activities are result of object interactions. These interactions are usually spatial relationships such as touching, disconnected etc. For example for a person throwing a ball, the activity involves events such as picking the ball and throwing the ball. Objects are ball and person and spatial relations are that person in contact with ball, then disconnected from ball as it moves away and so on. If these various properties of an activity are represented in an abstracted way then a good representation is achieved.

Sridhar [16] proposes the idea of qualitative spatio-temporal graphs to model activities. There has been little work previously seen with this graphical representation of activities. Hamid [4] uses a similar approach by making use of suffix trees for unsupervised activity analysis.

Sridhars approach works on analyzing a cluttered scene with noisy observations or noisy interactions. It is proposed that not every interaction in a scene might be of significant value and there are some interesting interactions that need to be represented in order to represent the whole activity. An example of an interaction graph is shown in figure 2.3.

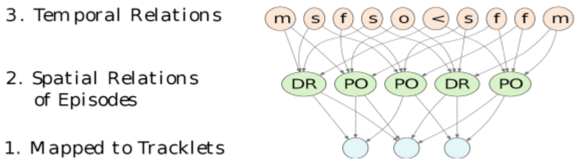


Figure 2.3: A sample Interaction Graph [Sridhar et al 2010]

The graph has three layers. The first layer represents the various tracks of objects through out the video, the second layer describes the spatial relation between pairs of track such as disconnected, overlapping etc. and the last layer defines the temporal relationship such as before, after etc. Intuitively all single events or sub set of events are simply sub-graphs of the interaction graph which are by them selves smaller interaction graphs. The model generated provides a probabilistic framework. Given a set of tracks from a certain video, the model that most likely interprets the activity is sought out. Using posterior priorities on each event class, the most likely model can be found. An interesting technique adapted in this approach to reduce the noisy or coincidental event graphs is by pointing out two properties that make a graph interesting [14]. It is said that interesting graphs or event graphs that represent some semantically meaningful information must be:

1) Maximally Frequent: If the event is semantically meaningful it is assumed to be more frequently seen in the activity graph than a noisy or coincidental event.

2) Sufficiently Interactive: An event is sufficiently interactive if the objects and their tracks in-

terweave with each other uniformly across the sequence. An observation where one of the tracks is completely off from the others is not considered sufficiently interactive and will there be regarded as a noisy event.
This approach filters out candidate event graphs that are noisy and coincidental with reasonably good accuracy.

To train the event class model, each interaction graph generated from the video is annotated with their class labels and learning is performed on this. There may be likely instances that a single interaction graph has multiple class labels of activities that can be assigned to it since it is modeling many activities taking place. This brings the need of a multi-label learning system. Also each label may have overlapping graphs associated to it; therefore a bag of graphs is created for learning process.

To classify a newly acquired video to an activity label is a challenging task. Since the activities are modeled as graphs, there is need for a way to measure similarity between graphs from the models and graphs from the newly acquire test data. The task can be formulated in the following way. We are tying to find the most probable match of interaction graph that is made up of many sub-interaction graphs using the learned model. The process can be thought of as matching interaction sub-graphs and finding the overlap between them. Through out the video, various time intervals are labeled with a verb such as loading trolley, picking bag etc, based on the most probable sub-graph match for that interval.

This approach proposes a complex but highly representative method for activity analysis. It is fairly robust to noise and coincidence data. The results from experimentation show a decent accuracy of activity classification for a dataset that is challenging. More recently research is being done on supervised activity analysis using this approach.

2.4 Relation to our work

The novel aspect of our work is in interpret video activities in terms of a hierarchical and logical event model that uses a multiple hypothesis framework to resolve ambiguities in observation. Through literature review, we have seen that the models that are most similar to our proposed system is the logical model and the hierarchical hidden markov model. These two models are well known for event classification using probabilistic analysis.

However, we put forth the hypothesis that the higher level structure for most activities are on the whole known in advance. For example, there is usually a standard way to lay a table. A more complex example would be, that once a type of dish is known in advance, the procedure at the higher levels largely gets fixed. A third example baking a cake or baking a pizza would have high level activities of 'baking' always the same. The changes would happen in the arguments. Hence, it may be overkill to model too much flexibility at the higher levels and pay the cost of extensive training data and complex training and inference schemes, that are characteristic of HMMs. Such HMMs become even more complex when they are extended with logic. While they are designed to handle variance in events, they do not naturally provide a way to encode constraints in the form of common sense rules that may be used to handle ambiguities in observation.

Our novel approach incorporates hierarhchy and logic. While we do not focussing on modelling event variablity, we focus on the problem of interpretting ambiguity in observations using a *multiple hypothesis approach*. The logical hierarchical event model encodes constraints with regard to action transitions and the passing of objects from one action to another, and are used to resolve ambiguities in observation.

We believe that our approach can be potentially developed into a top down bottom up framework, where observations about some objects and actions can be used to infer what kind of activity is going on i.e. to infer and fix the script, for example which dish is going to be prepared. This fixed script can then be used to interpret the video in terms of actions and sub-actions applied to different types of objects. We believe that this work is a starting point of such a promising approach towards activity analysis.

2.5 Conclusion

We have described a number of approaches for activity analysis. We started with basic simple approaches that made use of low level features such as background subtracted blobs for template matching and optical flow such as STIP features. These were good for simple activity recognition with single objects or subjects. We then described state-space approaches that are probabilistic approaches for more complex scenes with multiple actions and multiple objects. Lastly, we mention some approaches that are recent innovative research. These new approaches can be thought of as hybrid approaches as they make use of multiple techniques that were previously described. The new approaches create high-level features and representation that help with detection and analysis of videos with a much higher level of complexity. It can be observed that through time activity analysis has moved from simple single subject activity analysis to complex multi-subject multi-activity analysis. This is still a very open research area and it can be expected to see improved techniques in the future.

Chapter 3

Logical Hierarchical Event Modelling

3.1 Introduction

One aspect of human activities is they tend to be naturally hierarchical. Thus, a hierarchical representation is natural in order to represent human activities in terms of actions and their sub-actions. An example of human activity expressed in the form of an hierarchy is shown in figure 3.1. From this example, it can be seen that even a simple activity such as making salad, comprises of actions such as prepare vegetables, make dressing etc. This hierarchy represents the structure as the sequence of actions. It can therefore, be used as a generative model to simulate such a sequence. In figure 3.1, an example sequence of actions would look like *wash, chop, peel* etc.

Another aspect of human activities is that they arise when actions are applied to objects. For example, a action sequence such as *wash, peel, chop* could be enriched with object information so that we may have *wash carrot, chop carrot, peel carrot* then *wash onion, peel onion, chop carrot* etc. Thus actions may be naturally represented by logical predicates so that the objects constitute arguments for these predicates.

Motivated by these two considerations, we propose a Logical Hierarchical Event model for modelling human activities in this chapter. In the following section, we formally describe our proposal as a generative model that extends a hierarchical event model with logic. In section 3.3 we describe how such a model can be used for interpreting activities in terms of a hierarchical event structure. Section 3.4 describes how this model can be learned from data. The final section 3.5 describes the spatial representation used in the proposed event model.

3.2 Formulation

A logical hierarchical event model is defined as the following 5-tuple : $S = \{\mathfrak{E}, \Sigma_v, \Sigma_n, \mathfrak{X}, \mathfrak{H}\}$. This structure is illustrated in figure 3.2 and described below.

1. $\mathfrak{E} = \{E_1, E_2, ..., E_n, ..., E_N\}$ where N is the total number of nodes/states as shown in figure 3.2.

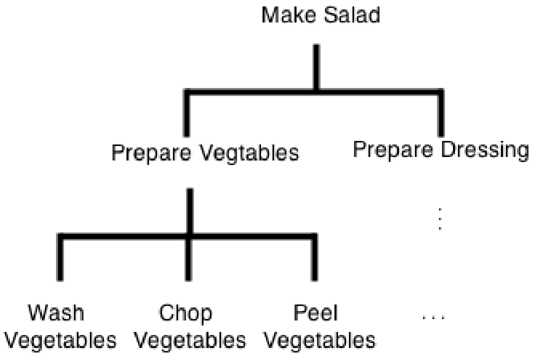


Figure 3.1: A common activity represented as a hierarchy of events and sub-events

Each of the nodes represent a single event. Events are defined as a combination of actions and objects. Notice that they are presented on different levels l.

2. Σ_v: defines a action state transition matrix that describes the allowed action transitions between states. These are shown as red arrows in figure 3.2. Note also that some action transitions are repeating and those edges represent self-transitions. We further explain this matrix below.

3. Σ_n: defines a object state transition matrix that describes the allowed object transitions between states. These are shown as blue arrows in figure 3.2. Similar to action transition, object transition can also be seen as repeating and self-transitioning edges are present. We further explain this matrix below.

4. $\mathfrak{X} = \{X_1, X_2, ..., X_m, ..., X_M\}$ where M is the total number of objects in the scene. This is seen as an arguments passed in at every event node E_n as presented in the hierarchy in figure 3.2.

5. $\mathfrak{H} = \{H_1, H_2, ..., H_o, ..., H_O\}$ where O is the total number of interacting agents in the scene. In figure 3.2, these are also presented as an argument passed into the event nodes.

Action transitions and object transitions are the two types of transitions that can occur within the hierarchical structure. These are described below.

Action transition: The first type of transition is action transition and are represented in a two dimensional matrix $\Sigma_v = \{a_{ij}\}$ where a_{ij} is 1 or 0 depending on whether there is a transition between i^{th} event and the j^{th} event nodes. Action transition involves actions holding for a certain duration. However for the bottom layer of the hierarchy, these durations correspond to the unit time step (in the case of videos - one frame).

A special case is when the transition could be made onto the node itself. This is repetition of the same action for the same objects. The objects do not change during the transitions. These transition carry the same objects with them. An example of a hierarchical state machine with just action transition is shown in figure 3.3. It is clearly seen, from the example, that the self-transition or repetition

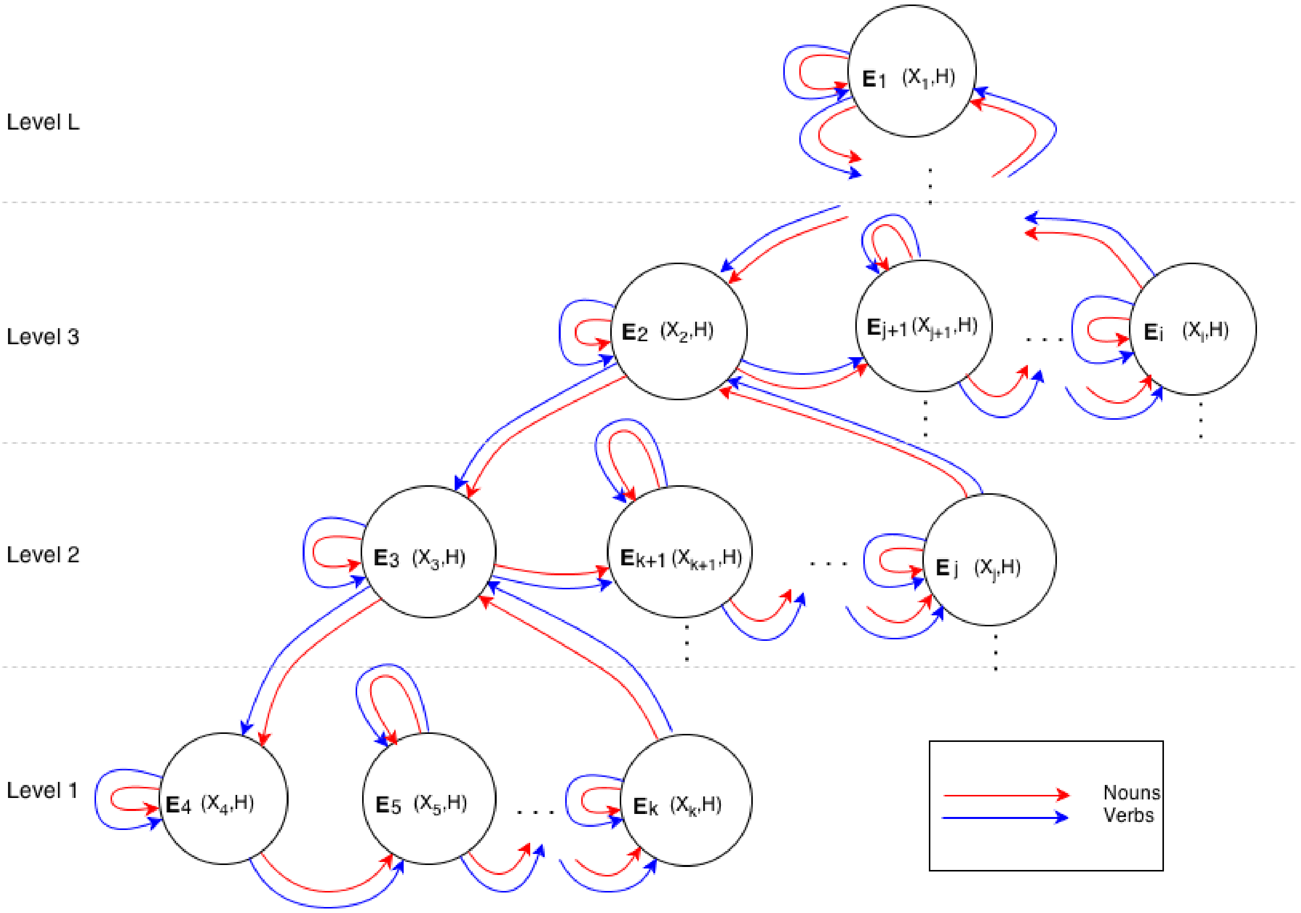


Figure 3.2: A hierarchical state-machine structure showing the general framework of the proposed solution. The nodes are numbered in a pre-order traversal way. E denotes the events. X and H denote the arguments passed where X are the objects in the scene whereas H are the interacting agents. The red lines show object transitions where object arguments vary while the arguments stay the same.

occurs at the 'chop' action whereas the object 'cucumber' remains the same. The sequence of events generated here would be as such: chop cucumber, chop cucumber, chop cucumber. This would repeat for a certain duration. These transitions indicate events that are repetitive whilst interacting with the same object.

Object transition: The second type of self-transition is object transition. Object transitions are stored in three dimensional matrix $\Sigma_n = \{b_{ijk}\}$, where the i_{th} row and j_{th} column represent the transition between the i'th and j'th action E_i and E_j. The k_{th} index describes which arguments would be passed from E_i to E_j and also in what order they are passed on. In other words, $b_{ijk} = 1$ means that the objects X_{ik} is first passed on from E_i to E_j. More than one object may be passed on at the same time from an event to another. However, if $b_{ijk} = 2$ this means that the objects X_{ik} is first passed when the transition from event E_i to E_j, happens for the second time.

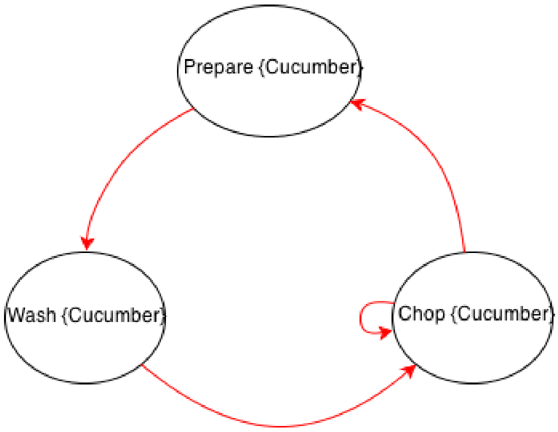


Figure 3.3: A simple instance of the hierarchical state machine model showing action repetitions

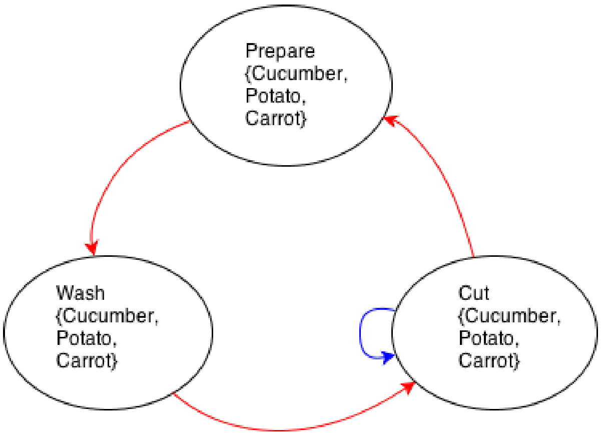

Prepare {Cucumber, Potato, Carrot}
Wash {Cucumber, Potato, Carrot}
Cut {Cucumber, Potato, Carrot}

Figure 3.4: A simple instance of the hierarchical state machine model showing object repetitions

Object transition involves a transition from event node to another where a different set of object arguments are passed. Similar to action transition, object transition can be repeated on some nodes. This can be seen in figure 3.4. Figure 3.4 illustrates this using an example. Here the object is being repeated. Multiple objects are passed in as arguments and at the 'Cut' action, the repetition that takes place would generate the same action with the various objects. A sample generated event sequence at the 'Cut' node would be as such: *Cut Cucumber, Cut Potato, Cut carrot*. Each time a Object transition takes place from one node to another, with a certain set of arguments i.e. objects, these are passed on to the child nodes. The two types of transitions can occur on the same node as seen in nodes in figure 3.2. This is where the event would be repeated by objects and for each object, it may be repeated by the action.

The model described above provides a very generalized structure applicable to many domains. Therefore an instance of the model is derived according to the specific domain requirements. The instance of the model used in our implementation is shown in figure **??**specMod).

3.3 Event Detection

In the following, we describe how the logical hierarchical event model is used to detect events in video. The above section described the hierarchical logical event model as a generative model that is able to generate a sequence of events that comprise the activity. This is primarily a top-down approach. Event detection is carried out when this generative model is used to detect incoming real time observations which are passed bottom up. In other words, the model interprets the observations from the video.

The observations obtained from video are in the form of bounding boxes of object tracks. These observations are used to infer events from the lowest level of the hierarchy to the highest level. The observations first trigger the events at the lowest level. When the sequence of events that constitutes the children of a higher level event is completed, the higher level event is triggered. This process continues till the event at the highest level of the hierarchy is triggered. This process constitutes the hierarchical event detection. During the process, the objects associated with each event is passed on to the higher levels, thus populating the hierarchy with triggered events and their corresponding objects at each time step. In this manner, the Logical Hierarchical event model is used to interpret the observations that constitute the activities captured in the form of a video.

3.3.1 Multiple Hypothesis Event Detection

One of the key features of the proposed framework is it's ability to resolve ambiguities in the observations using a multiple hypothesis approach. Ambiguity in interpreting observations as events may arise when more than one event sequence could be hypothesized. For example, when an object may be taking in the presence of other objects, it may appear that the agent is taking more than one object. Thus multiple hypothesis about which objects are taken can arise. However, such ambiguity may be resolved by waiting for a little while, when the agent starts carrying the object he has really taken as this evidence can be used to resolve the ambiguity that was produced earlier. Thus future events can be used to resolve ambiguity about past events.

The resolution of such ambiguities is facilitated by constraints such as *if an agent carries an object after taking, it is the same object that is being taken.* The proposed logical hierarchical event model naturally handles such constraints in the form of the object transition matrix Σ_n described above, as this matrix specifies which objects can be transitioned from one state to another.

More generally, the multiple hypothesis approach produces several hypotheses at each event state when ambiguity is present, and keeps a memory of such hypothesis in a stack. Then it uses constraints to resolve these hypothesis at a future state. We discuss the two types of constraints corresponding to the two types of transitions that can take place in the proposed event model. **Object Constraint.**

An object constraint is given by the object transition matrix Σ_n where Σ^n_{ijk} E_i and E_j, which specifies which objects indexed by k in X_i are required to be passed to X_j depending on whether the E_i and E_j are connected $\Sigma^v_{ij} = 1$ and whether Σ^n_{ijk} is greater than 1 or 0.

Note that we require all the objects in the children event nodes to be elements of the objects corresponding to the parent node. However, generally speaking there are no such constraints on event nodes of the same level may. These constraints between nodes of the same level may be introduced according to the requirements of a specific domain. For example, if there is a transition from *take* node to *carry* node, we may require all the objects in the latter node to be present in the former thus expressing the constraint that those objects that are carried have only been taken.

More generally, we have a set of hypotheses H_i at each event node E_i. The generation of hypothesis depends on the groups of objects X_i. A hypothesis stack can be defined $H_i = \{H_i^1, H_i^2, ..., H_i^R\}$ where R is the total number of hypothesis. Hypothesis updates occur when there is a mismatch of the objects X that are passed in at two consecutive event nodes on the same level l. This indicates that an event node did not recognize some of the argument objects. The hypothesis stack is then updated to remove all hypothesis that contain the unrecognised objects.

Action Constraint. The action constraint places constraints on the transitions between one action and another. It also typically arises in order to avoid potential indefinite repetition of a action in one of the states by enforcing an upper and lower limit to the number of times a single event can appear before switching to the next in order to recognize a higher level event. An example of how such a constraint is useful is as follows. When an agent take an object, he is initially disconnected and then connected to the object. Then he starts carrying the object. However, both taking and carrying involve being connected to the object and there is an inherent ambiguity as to when taking stops and carrying occurs. This can be resolved in the future by using the evidence that he has indeed completed the task of taking and has started turning and walking towards another destination. A constraint on the maximum duration which taking can occur also can be used as a constraint to resolve such ambiguities. Thus the multiple hypothesis approach naturally fits in with action constraints to resolve ambiguities in the past by using evidence from the future.

3.4 Learning the Event Model

The proposed event model can be easily hand coded for most domains. However, it can also be learned automatically from video data if this data was annotated with events at different levels in the hierarchy. In other words, our ground truth consists of event labels at each frame. Since there are multiple levels in the activity hierarchy, the ground truth consists of labels for each of the levels of activity. Table 3.1 shows a sample ground truth table.

In order to learn a hierarchical structure, we proceed by learning layer by layer of the hierarchy, starting from the bottom most layer and proceeding upwards. Using the ground truth table, we mine the events transitions that define the relationship between the same and different levels of event in the activity hierarchy. We start with lowest layer and try to learn the most commonly occurring combination/pattern of the low-level events that correspond to each of the events in the immediate higher level. Once this is learned, we repeat the process for each higher level.

For example, in table 3.1, we first to try and learn the events derived from the first level. To do this,

Frame	Level 1	Level 2	Level 3	Level N
1	DC	TakeTray	SetTray	...
2	DC	TakeTray	SetTray	...
3	PO	TakeTray	SetTray	...
4	PO	TakeTray	SetTray	...
5	PO	PutTray	SetTray	...
6	DC	PutTray	SetTray	...
7	DC	TakeCup	SetCup	...
8	DC	TakeCup	SetCup	...
9	DC	TakeCup	SetCup	...
10	DC	TakeCup	SetCup	...
11	PO	TakeCup	SetCup	...
.	.	.	.	...
.	.	.	.	...
.	.	.	.	...

Table 3.1: Ground Truth of a Sample video in Kitchen Dataset

the actions are observed in level 2. For each action in level 2, the corresponding action change in level 1 is calculated. Considering frame 1 to 4 the action TakeTray in level 2 holds, the corresponding action change in level 1 can be seen to move from DC to PO. That window is labelled as DC-PO. Similarly, the entire list is observed and Take windows will be labelled as their corresponding relation. After which the list of calculated action rules for Take are checked for most occurring rule. This would be finalized as the label for Take. Having learned the rule for an event such as a take, the procedure checks for the object type that is involved. If for example, tray is involved then a rule such as *Take Tray* is learned. A similar logic is learned to generate the hierarchical event structure. This structure is reflected in the action state transition matrix described above.

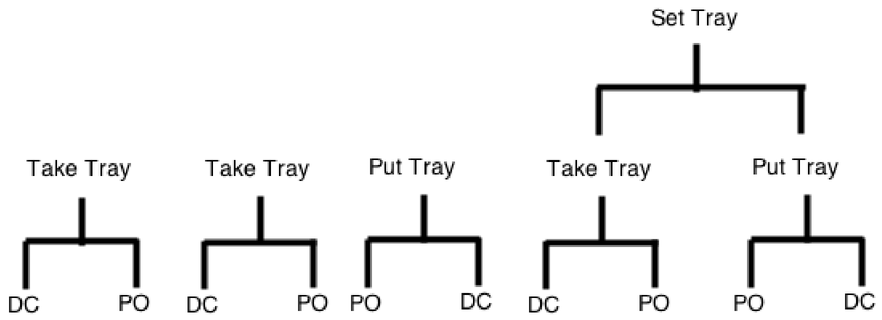
When an action is involved with more than one object, the noun state transition matrix ensures the order in which the objects are passed on. For example, if Take is repeated with an tray for the first time and with a cup for the second time, the learned index for tray is 1 and for the cup is 2.

Using the above learning procedure, the action and object transition matrices are updated across all the videos in the training set. Thus we learn the logical hierarchical event model, that can be used to detect events using the approach described in the previous section.

After a single iteration, all events relations for one level are learned as seen in the hierarchy diagram in figure 3.5(a). Similar technique would be applied to learn the rules at higher levels until level N. The evolution of the hierarchy can be seen in figure 3.5(c)

3.5 Representation of Interactions

Representation entails modelling the relationship between objects and the subject in a scene. In a typical scene with an activity, a subject (which could be a human, a robot etc.) is expected to interact



(a) Level 1 (b) Level 2 (c) Level 3

Figure 3.5: Evolution of the hierarchy at each level learned

with a single or multiple object (table, chair, floor, utensil etc.) to perform an activity. The relationship between the subjects body part and any one of the object could be defined in terms of space or time or both. Spatial relationships define the physical space features, such as distance, between the objects.

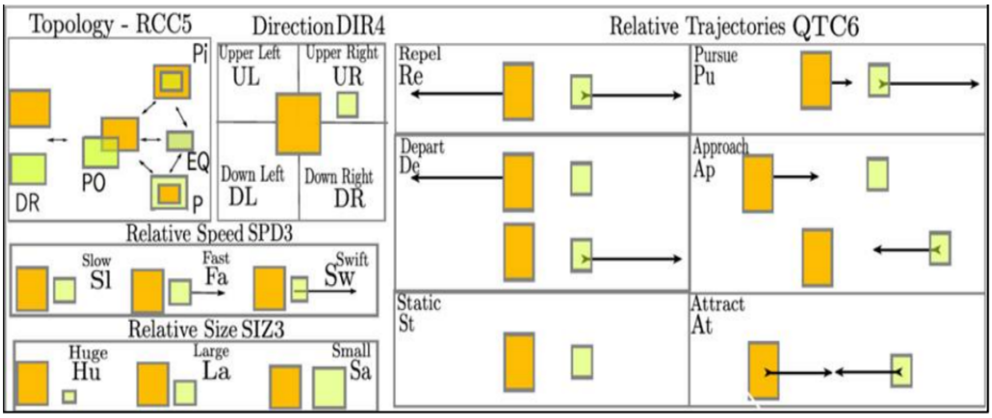


Figure 3.6: The five different qualitative spatial relationship categories, Topology (RCC-5), Direction (DIR4), Relative speed (SPD3), Relative size (SIZ3) and Relative Trajectories (QTC6) (Sridhar et al., 2011)

In our proposed solution, to model the interaction between the various objects and the subject in the scene, we used a simplified version of the topological RCC-5 Cohn [1] representation, figure 3.6. This makes use of identifying the major objects in the scene along with the interacting subject, and observing the changes in the spatial interaction between the two. There are however many different relationships such as PPi, PP etc. Sridhar in [2] simplifies the RCC-5 calculus by eradicating the EQ relation. This is done on the basis that this relationship is unlikely to occur in real video data sets. Moreover, the PPi and PP relations are combined in a single relation P. This is done because in our representation both these are interchangeable. This produces a simpler set of relationships.

Chapter 4

Experiments

4.1 Introduction

In this chapter, we evaluate our framework described in Chapter 3. Experiments will be carried out for activity recognition where the results would be compared against the ground truth. Our experiments will be evaluating the basic system, a system with the inclusion of constraints and finally a system that is machine learned. The dataset used in the experiments is described in the next section, followed by some of the specific implementation details and decisions that were made when implementing the system. This would be followed by evaluating various systems where in each the results and conclusions are presented.

4.2 Dataset

4.2.1 Introduction

The data set to be chosen for training and/or testing the proposed solution is a major part of any project. For our experiments we chose the TUM kitchen [17] data set. This data set provides a challenging set of videos where many different activity recognition or object recognition tasks can be performed. The data set is described in a little more detailed next.

4.2.2 Motivation

The TUM kitchen dataset is a set of videos recorded in a kitchen scene. The dataset is primarily made to be used in areas of segmentation, activity recognition, tracking etc. The data set consists of a simple everyday actions in a kitchen which is setting up a table top. This involves placing a mat on a table, then a napkin, a plate and a glass on top of the mat and finally a fork, a knife and a spoon on the sides of the plate. The purpose of this dataset is to provide a challenging dataset to researchers that wish to perform various activity recognition and analysis tasks. Previous data sets used for such analysis have always contained well articulated and well separated motion such as jumping, running, etc. In real world, however, the actions are performed in a human-like manner as shown by the kitchen data set. The common actions performed in the dataset are reaching for an item, grabbing an item, carrying an item, placing item on a table, opening/closing cupboard/drawer etc. The overall activity performed in the dataset is setting up a table top with slight variations.

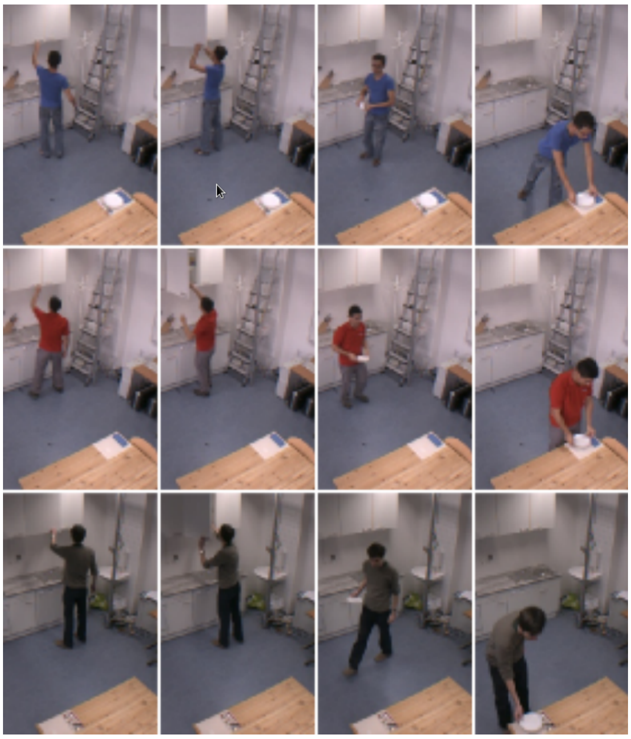


Figure 4.1: Example traces in the Tum Kitchen Dataset illustrating a person setting up a table

4.2.3 Data

There are 17 videos available where a person performs the ’setting up table top’ activity. Each video is recorded from four different camera angles. Each video is available in regular and high resolution. The videos primarily show the person setting up a table top in a robot like motion where items are placed one by one on the table. There are a few videos where the same activity is done in a more human like motion, and items are placed on the table in bulks.

The differences in the behaviour of the subject from video to video include changing the order in which the items are retrieved, changing the position of the items on the table top, changing the items themselves, changing the route of entry and exit from the room, introducing reluctance when grabbing an item and using switching between the use of left or right arm when manipulating objects.

There are a approximately 7 or 8 objects present in the scene. Only three of the objects have RFID tags on them. For our system, we are considering the use of only three of the objects namely the Tray, Napkin and the Plate. Other objects do not provide any additional variation in the main activity and thus are just repetitions.

Metadata

- Motion capture information: Motion capture data is available for the person in a Biovision Hierarchy format. The data captures position of 28 joints of the persons skeleton at each time frame. The data is available as absolute positions in a three dimensional space. In order to project these points onto the two dimensional camera scene, we used the provided camera extrinsic and intrinsic parameters. The following equations were used for 3D to 2D point conversion. Given a point (X,Y,Z) in absolute 3D space, we wish to find the corresponding point (u,v) in camera projection. We are given the extrinsic and intrinsic parameters of the camera.

  $$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = A[RT] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

  where A is the intrinsic parameters of the camera namely the focal length f and the principle point c These are presented in combined matrix of the form

  $$A = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

  R and T are the extrinsic parameters namely rotation and translation matrices. Once the world coordinates $[X_w, Y_w, Z_w]$ been converted to camera coordinates $[x, y, z]$, the following equations can be applied to get the 2D perspective projection of the points.

  $u = x/z$ and $v = y/z$

  We use the motion capture information of the right and left hands as interaction bounding boxes on the person.

- RFID tags: The RFID tags capture the position of only some of the items in the scene. These are tray, napkin, plate and cup. Only the start and end positions of items are available when they are moved. The remaining tracking information was either manually annotated or automatically approximated using the hand position from motion capture data.

- Ground Truth: The ground truth data labels each of the hands and the trunk of the body. Labels include basic events such as reaching, reaching up, grabbing, opening drawers etc. No item information is available. The ground truth data was updated to be compatible with our systems requirements. The following modifications were made.

  - Event Labels Grouping. The original ground truth used some very specific event labels. These were grouped together in order to obtain more general event labels. For example event labels such as Reaching and Reaching Up were grouped together to just Reaching.

- Object Information Addition. The original ground truth only had event labels for each frame that defined the event or action being performed but no information on the interacting objects. Our modification included ground truth to be updated to include object information along with the event label. For example Reaching changed to Reaching Plate at the times where that object was involved.
- High Level Events Addition. Original ground truth data only had basic low level events such as Reaching or Grabbing etc. Since the requirement for our proposed solution was the recognition and labelling of higher level events such as Setting, therefore the ground truth was updated to include all high level events as well.

4.3 Implemented system

4.3.1 Introduction

In this section, we would describe how the proposed framework was implemented to work with our specific requirements. The proposed framework provided a flexible model that can be applied to any problem domain. Application of the model required identification of events at each of the levels, how they relate to each other and how can they can be connected with each other. We also would require to identify the individual interacting objects in the scene in our case,for example, it is the hand and any one of the items in the scene. After this domain knowledge is acquired, the model can be specified to the needs of the problem domain.The proposed framework in section 3 was used in the following manner:

4.3.2 Specific Framework

The specific framework is constructed by observing the events found in out dataset and representing the entire activity as a hierarchy. The specific system model can be seen in figure 4.2

Figure 4.2 comprises of four levels. This hierarchical structure is a natural way of visualizing any activity. Since a high level activity comprises of a number of low lever events, a hierarchy could be built to model entire activities. An example could be given for an activity of 'Setting up table top'. The overall activity is making coffee which comprises of sub events such as 'Setting items' etc. These events can further be broken down into lower level events such as 'take tray', 'take spoon' etc. There might be a possibility to further break the events, but the requirement for this model is to reach a sufficiently low level that can be easily defined in terms of spatial relations. These can then be modelled into the different layers of the hierarchy structure. All layers are described as follows.

- Spatial Events: This is the lowest level of the hierarchy. Spatial events are the spatial relationships captured from the actual video data. These are relationships between any two objects in the scene. In our dataset, the two objects are usually the primary hand and an object on the counter. The relationships can be any of the relationships in R. They are repeatable because

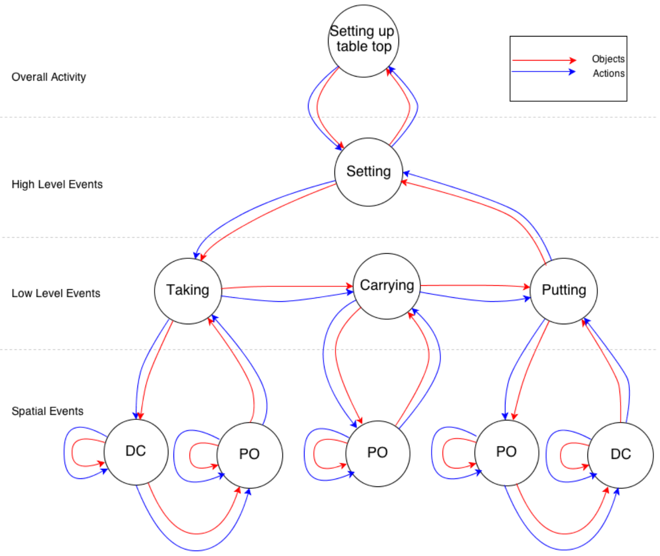


Figure 4.2: Specific model to recognize activities in the Tum Kitchen Dataset

they can stay the same over a number of frames. An example of this would be when the object are not in contact, the relationship would be DC DC DC and so on. This signifies disconnected relation for the number of frames.

A disconnected relation DC can move to a partially overlapping relation PO. The partially overlapping relation can also be repeated for a number of frames. However, this state exits to a state in the level above firing a low level event.

- Low Level Events: The Low level events are fired when there is a change in the spatial events. A series of spatial events are obtained from the lowest level in the hierarchy. Based on predefined rules, low level events are inferred given the spatial event chain over some duration in a video. As seen in figure 4.2 a chain such as DC DC DC followed by PO PO will fire an event Taking in our kitchen dataset.

  A different combination of spatial events would fire different low level events and a series of low level events would then fire a higher level event.

- Higher Level Events: Moving up the hierarchy, the high level events are events that comprise of many low level events. They provide a higher level of interpretation of the overall activity happening. Similar to low level events, high level events are inferred using predefined rules. However at this level, the rules are defined using the previously inferred low level events and not the spatial events. As seen in figure 4.2, a series of other low level events followed by a series of another low level events comprises of the high level events. This means a Taking event following by a putting events are would infer a high level event, setting, in our kitchen dataset.

- Overall Activity The highest level of the hierarchy defines the overall activity in terms of its sub events. The overall activity is recognized based on the series of the higher level events inferred before. In our kitchen dataset, there is only one type of higher level event which is setting. these repeat multiple times indicate the setting of multiple items and thereby inferring the main activity as Setting table top.

4.3.3 Rules and Constraints

4.3.3.1 Rules

The dataset comprises of a general action of setting up a table top. This involves retrieving various items from the counter and placing them on the table. After observation of the dataset, it was decided to break down the general activity of setting up table top into high level events of setting up items. These were further broken down into low level events of taking, carrying and putting. These events could now be modelled as spatial relationship changes between the hand and objects in the scene at certain frames.

Rules are defined to model events from spatial relationships. These events are taking, carrying and putting. All higher level events are inferred from the events on the immediate lower level only. The list of rules defined are listed next.

- Taking : If the spatial relation between the primary hand and any object changes from disconnected DC to partially overlapping, that event is labelled as taking since this simulates the hand moving in for the object.

- Carrying: If the spatial relation between the primary hand and any object sustains as partially overlapping PO, that event is labelled as carrying since this simulates a carry verb.

- Putting: If the spatial relation between the primary hand and the object changes from partially overlapping PO to disconnected DC, that event is labelled as putting since that simulates a hand moving away from the object hence putting.

- Setting: On a higher level than previously described events, we have setting. This is fired when there is a succession of taking, carrying and putting low level events. This defines that an item must have been set on the table.

- Setting up table top: If there are successive setting events this mean that multiple items have been set on the table and therefore the event must be setting up table top.

4.3.3.2 Constraints

The system that we proposed has the capability to be constrained. The main constraints we implemented on the system was multiple hypothesis and duration. These constraints define what is allowed whilst the model is used. The duration constraint simply defines the number of time a repetition is allowed in the spatial nodes before it is required to move to another node. Spatial node repetition could not be done infinitely. Multiple hypothesis constraint defines the allowance of many hypothesis to be formed and how to resolve them. following is a description of how multiple hypothesis are resolved.

Hypothesis resolution There are a number of ways in which a hypothesis can be resolved. The following list presents the ways employed in the system for hypothesis resolution so far.

1. Object-Subject Distance measure: For hypothesis that are multiple taking events as in the case of reaching for a number of objects that are in closed proximity, a distance measure between the subject and each of the involved objects is recorded. After a threshold number of frames, the distance reading are put through a simple smoothing function as follows:-

   $sd_t = \sum^i d_i/(n+1)$ where $(i = t-n,...,t+n)$ and sd is the smooth distance, d is the original distance and i the smoothing range. n is typically set to 5.

   The gradient is then also threshold to identify distance that remained the same and ones that increased. This provides a true measure of which objects were carried as their subject-object distance changes would be minimal where as the objects that were left would have large subject-object distance changes. The hypothesis stack is then updated by confirming some and disproving others.

2. Persistence: At any point in time, if there is a change in temporal relation such as a DC becoming a PO at that frame, that event fired is set as a hypothesis. It is only confirmed if the newly set spatial relation holds for the following threshold number of frames. This minimizes the effect of any noisy jitter in the tracking of object causing the noisy reading.

3. Proximity to Destination: Finally, the destination of object track could either be manually labelled or clustered given object tracks through all videos. This destination can then be checked every time an object is disconnected from the subject. This ensures that the verb putting is only fired when the object is left to its final position and not when it is left else where and picked again.

   More resolution methods can be employed to make the system better equipped to specific domain requirements.

4.3.4 Conclusion

We have described the implementation of the event model here. The model is now tested to evaluate its performance against ground truth.

4.4 Evaluation Method

Our evaluation method primarily relies on the ground truth. The ground truth is readily available along with the data. We would compare ground truth to the observations using an overlap measure. This defines the amount of overlap a single event observation has with the ground truth. The percentage of the overlap could be threshold to label those instances as true positive.

Along with the overlap percentage, we would use a confusion matrix to check which events are getting confused with which other ones. F1 score that combine precision and recall, would also be presented to give a numeric measure of the quality of performance.

4.5 Exp 1: Performance of the Logical Hierarchical System

4.5.1 Introduction

In this experiment, we are going to perform simple evaluation techniques on the Logical Hierarchical system. The labels of verbs obtained from the logical hierarchical system will be matched against the labels obtained from the ground truth data.

4.5.2 Experiment

The most basic requirement to judge and evaluate a recognition system is through the use of ground truth. To do this, we require manual labelling of the videos that define truthfully which action happens at which point. Given this truth data, the evaluation of an activity recognition system simply becomes a matter of comparison of obtained label versus the true labels. The similarity between the two labelling shows the degree of accuracy of the system.

In this experiment, we are evaluating the performance of the system in its most simplest form. At this point, only the verb information is applied. No object detail is included in. The original ground truth would be used for evaluation and comparison because it also has no object information. The output obtained from the basic logical hierarchical system comprises of frame by frame labelling of the events detected. Each frame may be labelled with a single events or a number of events that include high level events as well. For example, frames where a taking, carrying and putting actions are happening are also where the higher level parent action of setting is happening. Therefore we need to incorporate all levels in the evaluation process.

To compare this hierarchical structure with the ground truth, we are simply going to add the higher level events/actions as more verbs at the end of the list of existing verbs. These would be compared with the corresponding higher level events/actions in the ground truth data. This way our total verb list produced in this experiment equals to the total number of events at all levels in the hierarchy. There would not be loss of information or structure since the comparison of each verb will be done independently. The results from this experiment are shown next.

4.5.3 Result

Figure 4.3 shows the confusion matrix obtained comparing the labels across all chosen videos in the dataset. The actual numerical matrix along with a pictorial representation is given.

	Idle	Taking	Carrying	Putting	Setting	Setting up table top
Idle	1601	0	0	103	93	13
Taking	211	158	97	9	0	0
Carrying	13	2	989	63	0	0
Putting	71	0	428	165	0	0
Setting	304	0	0	0	1889	0
Setting up table top	629	0	0	0	0	2250

Table 4.1: Confusion Matrix of testing simple logical Hierarchical system

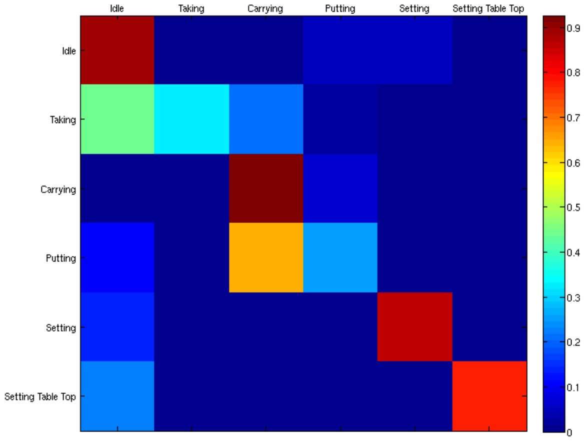


Figure 4.3: Normalized Visual Confusion Matrix of testing simple logical hierarchical system. F1 Score : 0.3386

4.5.4 Conclusion

A number of conclusions can be derived from this experiment. First of all, it can be seen that the system is generally performing well, as most high numbers lie in the general diagonal area of the confusion matrices. This means that activities are somewhat correctly recognized. The interesting observation here is the accuracy of the high level activities such as setting and setting table top. These are infered from low level activities using the model and not recognized from true vision data. The minimal amount of confusion for those activities shows good performance of the model.

Another observation is that major confusion lies in low level events that hold for a short period of time. These are events such as Taking, Carrying or Putting. Carrying holds for a longer period of time and therefore, atleast some overlap is bound to be recognized between the Carrying window in the observed data and the ground truth. Taking and Putting, on the other hand, suffer from their short interval in the recognition. Since these events are recognized solely by the spatial relation changes, they are usually lost within the ground truth where they are labelled much earlier than the recognitions.

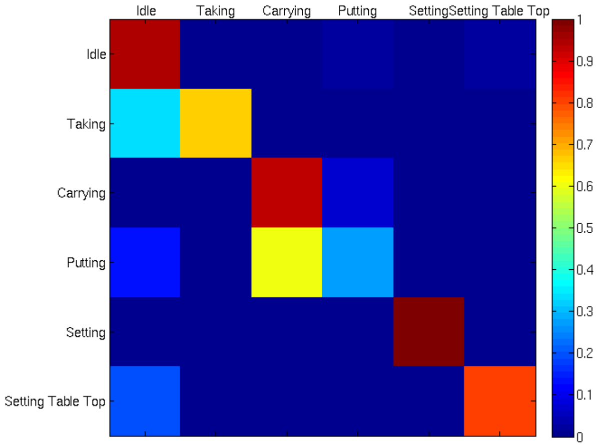


Figure 4.4: Normalized Visual Confusion Matrix of testing simple logical hierarchical system using Middle point of an event window and checking corresponding ground truth element

4.6 Exp 2: Logical Hierarchical System with added window size

4.6.1 Experiment

The second experiment is devised after the system has been altered to include the following two rules:

- Taking actions should start earlier than the recognition point, i.e. the point of spatial change.
- Putting actions should also start a certain number of frames in advance. Putting is only detected when there is a disconnect amongst spatial relationships. This time is too late, that is why according to figure 4.5, putting is most confused with carrying which is the immediate preceding action.

The experiment is carried out on an updated version of the system where the taking and putting action windows have been adjusted to account for the artificial delay because of the spatial relation change. The new system is tested with the same data as previous and the results are given next.

4.6.2 Results

The confusion matrices obtained here are presenting the improvement in the system using the adjusted window size for the vulnerable events. This evident fromt he increase in the F1 score as compared to the previous matrix.

4.6.3 Conclusion

The confusion matrices in figure 4.5, show a significant improvement over the previous ones in figure 4.3. There is much less mixing generally for most previously confused events. Vulnerable events

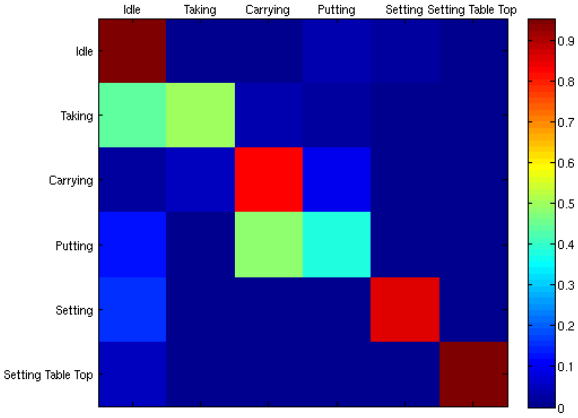


Figure 4.5: Visualized Confusion Matrix produced using the updated system for recognition. F1 Score : 0.3792

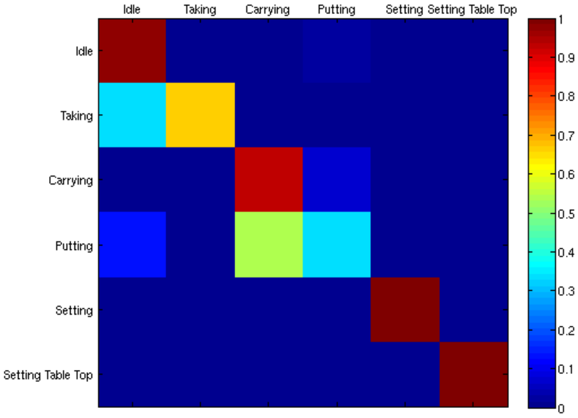


Figure 4.6: Normalized Visual Confusion Matrix obtained using the updated system for recognition

such as Taking and Putting still present a significant amount of confusion but it is an improvement over the system used in experiment 1. We would use this new updated system for the rest of the experimentation.

4.7 Exp 3: Performance of the Logical Hierarchical System with Object inclusion

4.7.1 Experiment

Object knowledge along with the verb describing the action included in our system. So far we have been experimenting with only action recognition. We are able to extend the system to include object information along with the action. For example, instead of recognizing events such as taking, putting etc. we could recognize event like taking tray, putting plate etc. The ground truth is again adjusted to this new requirement.

It is likely that we would have more false positives when incorporating the object information in the system. This is because the recognition becomes more specific to look for the object along with the verb. Previously an action could have many different object along with it. All those instances were recognized as true positives since only the action was considered. With the object information attached, there is a higher rejection rate for the instances. We carry out the experiment by including object information in the labelling.

4.7.2 Results

Figures 4.7 and 4.9 are displaying all possible class labels, therefore a short hand is created to represent a class by its initial. Provided is the order in which they are presented on the axis.

Short Hand	Word
I	Idle
TT	Taking Tray
CT	Carrying Tray
PT	Putting Tray
TN	Taking Napkin
CN	Carrying Napkin
PN	Putting Napkin
TP	Taking Plate
CP	Carrying Plate
PP	Putting Plate
ST	Setting Tray
SN	Setting Napkin
SP	Setting Plate
STT	Setting up table top

Table 4.2: Confusion Matrix of testing simple logical Hierarchical system

4.7.3 Conclusion

A high proportion of the data is within the ideal diagonal of the matrix. With object information, the data is recognized better since there are now more classes than there were in the previous experiment. Clearly the low level verbs can still be seen as problematic areas in the figure 4.5. But the overall performance of the system is good.

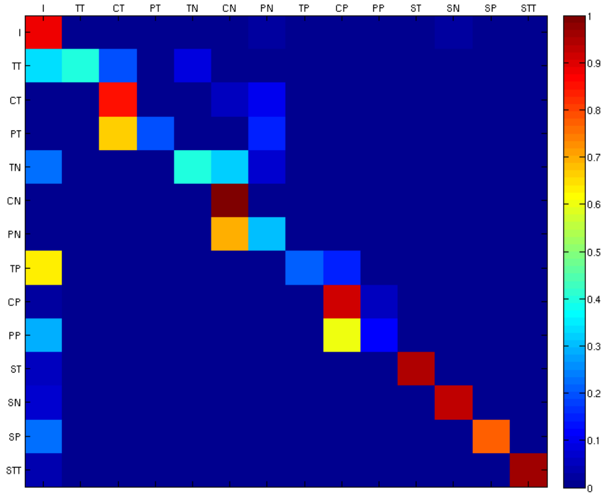


Figure 4.7: Visual Confusion Matrix for a video data with object knowledge.

4.8 Exp 4: Performance of Logical Hierarchical system with Noise

4.8.1 Experiment

The data used so far from the TUM Kitchen dataset is clean and repetitive. It does portray a real life scene but it is being carried out flawlessly in every video due to RFID tags. To check the system's performance to the fullest extent, we are required to introduce some noisy data into the system. In this experiment we are going to manually induce noisy readings in the system and then evaluate its performance. Following noise was added to the system:

- Multiple object interaction. The size of some object bounding box were increased to overlap other object's bounding boxes in the scene. This would create the simulation of a person reaching for multiple items but only taking one.
- Missing object track. Manual removal of some frames of object tracks from the scene to simulate noisy reading of data.

We will now run our program with the noisy data, the results the presented next.

4.8.2 Results

Figure 4.9 represents a pictorial confusion matrix showing a high number of falsely identified instances.

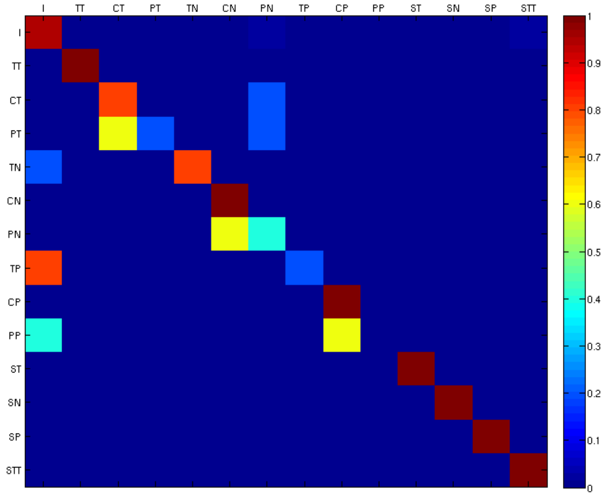


Figure 4.8: Visual Confusion Matrix for a video data with object knowledge.

4.8.3 Conclusion

Clearly there is a lot more mislabelled instances in Figure 4.7 than in figure 4.6. The addition of the above mentioned noise data has caused low level events to be even more misclassified. Also it is noticeable that some of the labels are not present. this is because no instances were found for them. In order to tackle this issue of noisy data or variations our of the ordinary, we introduce the multiple hypothesis system.

4.9 Exp 4: Performance of multiple hypothesis system with Noise

4.9.1 Experiment

An extension that is made to our system is the implementation of constraints. One such constraint is the multiple hypothesis system. This is where multiple occurrences of observations arising from noise or variations are resolved to select the true positives only while discarding the rest. Multiple observations are usually generated due to object detection errors. They could also be caused by confusing actions such as going to take two items at the same time etc.

In a non noisy domain, multiple hypothesis system produces the exact same result as the non multiple hypothesis one. This is because, there is no two events taking place at the same time and hence no ambiguity in which action is going on. The multiple hypothesis system requires ambiguity in order to hypothesise about the list of possible actions. We ran the experiment with the same data with multiple hypothesis system enabled. Results are presented next.

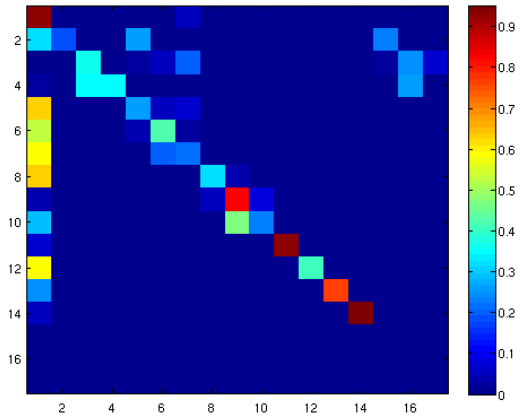


Figure 4.9: Visual Confusion Matrix for a video data with natural noise.

4.9.2 Results

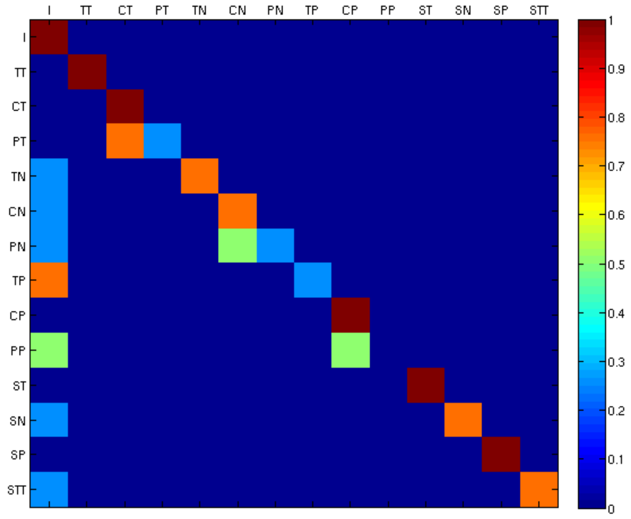


Figure 4.10: Normalized Visual Confusion Matrix for a video data with natural noise and multiple Hypothesis system.

4.9.3 Conclusion

The overall matrix has an improvement over the one presented in Figure 4.9. In a multi hypothesis system, it is likely that a single rule could not decide whether a particular hypothesis is valid or not. A number of rules are necessary for that. In our multiple hypothesis system, we use the distance measure between objects to detect if they are being carried or walked towards etc. Of course there are errors and false positives produced by the system. Improving on the multi hypothesis calculation for decision making would improve results. This is one of our aspiring requirements.

4.10 Exp 4: Performance of the Machine-Learned event Recognition

4.10.1 Experiment

A further extension to the system was made by implementing a basic machine learning system. This system was able to recognize the common occurring patterns in the ground truth and in effect produce the rules that define the activity hierarchy. In this experiment, the system will be given the ground truth data that is annotated and labelled to the three levels of hierarchy. The ground truth data is obtained from three of the five fully annotated videos. The system would then mine the most occurring rules. The rule generation is done in turns starting from the highest level of the hierarchy learning the immediate children nodes. Then for each children node, learn their children nodes and so on. Also in this experiment there is no object information involved and simple verbs are used. The results obtained from the experiment are displayed next.

4.10.2 Results

Rules Learned as follows:

- At Level 3:
  'Setting up table top' : 'SettingIdleSettingIdleSetting'
  'Idle' : 'Idle'

- At Level 2:
  'Setting' : 'TakingCarryingPutting'
  'Idle' : 'Idle'

- At Level 1:
  'Taking' : 'DCPO'
  'Carrying' : 'PO'
  'Putting' : 'PODC'

4.10.3 Conclusion

The rules mined from the ground truth look very similar to the one we manually programmed. In fact they are the same rules with just the addition of 'Idle', which can be discarded as it only fills the empty frames in the video. The experiment shows that the system is able to learn the rules and with high accuracy. The similar nature of the rules depends on the nature of the videos. Since every thing is quite similarly performed, it usually goes through the same sequence of events. Therefore the rules are of such similar nature to our system.

4.11 Exp 4: Performance of the Machine-Learned event Recognition with Object Information

4.11.1 Experiment

In the previous experiment, we only used the actions to learn the model's rules. Here we are going to use a further updated version of the ground truth that includes object labels as well as the action performed.

4.11.2 Result

- At Level 3:
  'Setting up table top' : 'SettingTrayIdleSettingNapkinIdleSettingPlate'
  'Idle' : 'Idle'

- At Level 2:
  'SettingNapkin : 'TakingNapkinCarryingNapkinPuttingNapkin'
  'SettingPlate : 'TakingPlateCarryingPlatePuttingPlate'
  'SettingTray : 'TakingTrayCarryingTrayPuttingTray'
  'Idle' : 'Idle'

- At Level 1:
  'TakingNapkin' : 'DCPO'
  'CarryingNapkin' : 'PO'
  'PuttingNapkin' : 'PODC'
  'TakingPlate' : 'DCPO'
  'CarryingPlate' : 'PO'
  'PuttingPlate' : 'PODC'
  'TakingTray' : 'DCPO'
  'CarryingTray' : 'PO'
  'PuttingTray' : 'PODC'

4.11.3 Conclusion

With the introduction of object information, a larger list of rules is produced. This model is expected to be more accurate since recognition will take place only when the action as well as the object is recognized. Higher level events would be generated not only by recognising a certain order of actions but also objects. The model becomes a little more constricted in the sense that if there is deviation from the defined order of objects, then no higher level events would be inferred.

The results are useful though to learn the most common or most popular method of performing activities. A robot would be able to learn that for example in a kitchen setting up table top, the sequence of events are to set a tray then a napkin then a plate. Compared to the previous experiment's model that teaches just three settings regardless of the objects involved are to be recognized as a setting up table top activity.

Chapter 5

Conclusion and Further Work

In this chapter, we describe the overall experience of the project. This includes evaluation of minimum requirements, have they been met, have the research questions been answered. What level of exceeding requirements were achieved. Limitations and challenges faced during the project would also be described.

5.1 Findings

Here, the conclusion drawn from the experiments are outlined from the previous chapter.

Experiment 1: Performance of Logical Hierarchical Event Model. This experiment showed that the model used to describe the activity performed with reasonably good accuracy. Performance of the system was measured against ground truth. This system was a basic system that implemented the event model to represent an activity in terms of its actions alone. The confusion matrix showed a high number of true positives. Low-level activities that had a short time span were noticed to be the most inaccurate of the recognitions. These activities were namely Taking and Putting.

Experiment 2: Performance of Logical Hierarchical Event Model with Low level event time span adjustment. The findings from Experiment 1 caused the motivation to implements additional rules on the system that improved its accuracy by a good factor. This experiment showed that low-level events such as Taking and Putting general are difficult to recognize from spatial event changes alone. Further rules were needed to provide new time spans that are realistic in defining such events. The result showed significant improvement for the Taking variable but an average improvement only for the Putting action.

Experiment 3: Performance of Logical Hierarchical Event Model with Object Knowledge: It was seen that incorporation of object knowledge in the model caused for better accuracy but lower recall. The recognized events were accurate but since a more specific pattern was now sought, there were less recognitions.

Experiment 4: Performance of Logical Hierarchical Event Model with Multiple Hypothesis: This experiment yielded important results. It was seen that recognition in a more challenging dataset, typically data obtained from vision techniques, is much prone to errors and noisy observations. The model was able to maintain a good accuracy for recognizing the events that were actually carried out while discarding the false recognitions. This is particularly useful in Real-time systems as many possible events recognitions could be hypothesized at any time. These can be run in parallel with the video and resolved at a later time when it is certain whether a hypothesis is true or false.

5.2 Project Evaluation

We would evaluate the project in this section. Given the various outlined requirements in Chapter 1, it would be possible to compare the achievements with the requirements.

5.2.1 Aims and Minimum Requirements

Minimum requirements and aims of the project were set in the beginning. Here we mention those requirements and assess if they were achieved.

1. Identification and further annotation of the TUM kitchen dataset videos to provide ground truth for experiments.

   This minimum requirement was completely met in section 4.2. The TUM kitchen dataset was thoroughly studied and modified to use with our systems. The data set originally had object-less annotation of the videos. The videos used in our project were completely annotated with:-

   - Object information: Along with the action, the interacting object information was also added.
   - Higher Level Events: The annotation only provided the basic low level events. Higher level events that comprised of the low level events were added for each frame in every video. Our model had three levels of high level events.
   - Simplification: Current video annotation provided a number of actions such as Reaching and Reaching up. Most of which were combined into simpler single events like Taking or Putting etc.

2. Creation of a Logical Hierarchical system capable of recognising simple events such as picking up some- thing or putting down something.

   Chapter 3 described a comprehensive model that could be used to create a Logical Hierarchical system specific to any domain. Section 4.3 described the implemented Logical Hierarchical system that was developed for the TUM kitchen dataset. The system was able to recognize simple low-level events such as taking or carrying based on the spatial relations that formed between the objects and the hand. Experiment 1 described the simple Logical Hierarchical system that recognized such low level events.

3. Creation of a system able to recognise complex events composed of simple events.

   The model described in chapter 3 and the Logical Hierarchical system implemented in chapter 5 both show a logical hierarchical structure that is able to recognize much complex activities and events that comprise of simpler low level events. Furthermore, it is not limited to one level of abstraction only. The system is able to recognize much higher level activities by piecing together the low level events taking place.

5.2.2 Research Questions

1. Could spatial relation be used to represent primitive interaction between objects? Spatial relations could represent interaction between objects with a good amount of accuracy. However, the specific event may have started before the spatial relation takes affect. For example when reaching something, it is manually recognized soon as the hand starts to move towards the object. However, in the spatial relationship concept, it will only take effect at the contact point between hand and object.

2. Could we incorporate and use action and object information together within a single recognition system. Yes, given our event model, the action and object information could be used together as arguments being passed from one event to the other. Activity recognition then depends on this passing.

3. Could we represent higher level events based on lower levels event and ultimately the entire activity as such a hierarchy? Yes, activities can naturally be represented in a hierarchical structure. In chapter 3, we have successfully represented the complex activity in as a set of sub events arranged in a hierarchy.

4. Can we use multiple hypothesis of events going on for better recognition. Multiple hypothesis aids in better recognition. Without multiple hypothesis system, multiple simultaneous detections could be confused with each other. With a multiple hypothesis system, hypothesis resolution criteria is used to only confirm what is actually happening.

5. Could we program rules that label various activity components in terms of the comprising events and spatial relations? Rule can be programmed to represent an event as a spatial relation change. Using domain knowledge we could say for example in a kitchen, the spatial relationship defining a take action is DCPO.

6. Can the model be learned automatically given ground truth data? The model could be learned automatically given ground truth data. It requires well annotated data and the higher levels the data is labelled, the more comprehensive model could be learn.

7. Would the automatic learned rules perform better than the manually programmed rules? Automatic learned rules system would perform better in theory, should their be noise in the data and

a lot of ground truth is available to train on. Our data didn't have any noise and little variation. Therefore our system was unable to learn anything different from the manually programmed.

5.2.3 Exceeding Requirements

The following exceeding requirements were also met.

1. First exceeding requirement is the use of multiple hypothesis system. Originally it was not proposed to be part of the system. But as the system progressed technically, it was evident that this would be needed to tackle many issues in a real human data.

2. Second exceeding requirement was the development of the machine learning system to learn the proposed model. The motivation for this was to have a full automatic system where an agent, computer or robot, be given the data and automatically a model is driven from it.

5.2.4 Conclusion

Overall we consider the project to be a success especially in the aspect of proposing a novel and new idea for a generative event model that make use of multiple hypothesis and parameter passing to model an activity with high accuracy. There were many other ideas explored such as spatial representation of the data, machine learning the model rules, programming a hierarchical structure etc. Performance of the model was satisfactory, there were misclassification of certain activities but all were at the low-level due to spatial relation changes. The activity that actually use the hierarchy was mostly recognized. It can be seen that the overview of the results show a consistent and decent result when recognizing an activity.

5.3 Further Work

There are a lot of potential research and experimentation that would improve the system. Following are the direction of possible future work.

1. We propose to improve the machine learning system by histogram comparison rather than simple data mining. Currently the system is finding pattern of most occurrence in the machine data as they occur. This may be accurate for a noise free dataset but it would be problematic for other datasets. A histogram comparison method would try to create a histogram model from the ground truth data. The models can then identified and recognize objects after training.

2. We also intend to use vision technique to identify the objects automatically. This would show a realistic measure of the system's performance with natural noise. This would produce a fully automated system.

3. Future work would also entail introducing more and more verbs in the system for recognition. This would allow for the system to recognize higher range of actions and cover the entire video rather than a small section of video.

Bibliography

[1] Ardhendu Behera, David C Hogg, and Anthony G Cohn. Egocentric activity monitoring and recovery. In *Computer Vision–ACCV 2012*, pages 519–532. Springer, 2013.

[2] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, 2002.

[3] Aaron Bobick and James Davis. Real-time recognition of activity using temporal templates. In *Applications of Computer Vision, 1996. WACV'96., Proceedings 3rd IEEE Workshop on*, pages 39–42. IEEE, 1996.

[4] Raffay Hamid, Siddhartha Maddi, Aaron Bobick, and M Essa. Structure from statistics-unsupervised activity analysis using suffix trees. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[5] Ivan Laptev and Tony Lindeberg. Interest point detection and scale selection in space-time. In *Scale Space Methods in Computer Vision*, pages 372–387. Springer, 2003.

[6] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[7] Gal Lavee, Ehud Rivlin, and Michael Rudzsky. Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 39(5):489–504, 2009.

[8] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[9] Justus H Piater, Stephane Richetto, and James L Crowley. Event-based activity analysis in live video using a generic object tracker. In *Third IEEE international workshop on performance evaluation of tracking and surveillance, Copenhagen*, pages 1–8. Citeseer, 2002.

[10] Ramprasad Polana and Randal Nelson. Low level recognition of human motion (or how to get your man without finding his body parts). In *Motion of Non-Rigid and Articulated Objects, 1994., Proceedings of the 1994 IEEE Workshop on*, pages 77–82. IEEE, 1994.

[11] MS Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1036–1043. IEEE, 2011.

[12] 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*, volume 3, pages 32–36. IEEE, 2004.

[13] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1470–1477. IEEE, 2003.

[14] Muralikrishna Sridhar, Anthony G Cohn, and David C Hogg. Relational graph mining for learning events from video. *STAIRS 2010*, pages 315–327, 2010.

[15] Muralikrishna Sridhar, Anthony G. Cohn, and David C. Hogg. Unsupervised learning of event classes from video. In *Proc. AAAI*, pages 1631–1638. AAAI Press, 2010.

[16] Muralikrishna Sridhar, Anthony G Cohn, and David C Hogg. Unsupervised learning of event classes from video. In *AAAI*, 2010.

[17] Moritz Tenorth, Jan Bandouch, and Michael Beetz. The tum kitchen data set of everyday manipulation activities for motion tracking and action recognition. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 1089–1096. IEEE, 2009.

[18] Pavan Turaga, Rama Chellappa, Venkatramana S Subrahmanian, and Octavian Udrea. Machine recognition of human activities: A survey. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1473–1488, 2008.

[19] Xiaogang Wang, Xiaoxu Ma, and W Eric L Grimson. Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(3):539–555, 2009.

[20] Junji Yamato, Jun Ohya, and Kenichiro Ishii. Recognizing human action in time-sequential images using hidden markov model. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pages 379–385. IEEE, 1992.

Appendix A

Personal Reflection

”There are no secrets to success. It is the result of preparation, hard work and learning from failure”. -Colin Powell

Undertaking any project of this magnitude was a daunting task for me. There were hurdles, problems, issues that constantly bombarded me. Ther were times when I found myself in extreme disturbance and distress. But looking back at the path that I took during the course of the project to get where I am now, I am honored for the hardships that befell. They have taught me a life lesson, no matter how deep you think you have sunk, givign up will only pull you further down. I am honored to have been supervised by such brilliant minds. The knowledge they have instill in me through the development of this project is immeasureable. This project has given me a new sence of hard work and determination. As I am now convinced that to achieve at your goal, the hardest path bears the best fruit.

While doing this project, I realized the endless possibilities that research work holds. It was often a struggle to block down onto a single idea when so many started to fly around, each looking more promising than the last. With the help of excellent supervision, it was possible for me to focus in just one direction. Even then I found myself at time to be confused or lost as to what to do.

The most challenging part in this project was mathematical formulation of the proposed structure. This is not a new phobia that I have, formulation has always haunted me and my stretagy had been simply to evade it. However, in this project, that was not an option. I am in great dept to my co-supervisor, who pushed me to tackle this problem on my own and iteratively reach the correct and proper formulation. It was then that I realized that nothing is hard if you put your mind to it. Infact, I enjoyed the expierience so much that I am eager to tackle a new problem and present a formal solution for it.

Challenges were faced when there was lack of documentation with some data used. It was a struggle to understand and parse data files to a workable form. I was forced to learn a brand new topic by myself to tackle this problem. Without any help or teaching, it took twice or three time longer to grasp the concepts. But that expierience was priceless as it taught me the importance of independence at this level of acedamia.

The time it took to write up too me by surprise. I was confident that the write up was soemthing that I was good at and could do fast, only to find out a major distinction between the write up of a scientific research and an essay. Learning this new method of writing was a rewarding expeirience. I highly advise future students to start the write up earliest as there is no time where the write up could finish as one expected.

I would advice future students to never take any aspcet of the project for granted. It is a huge undertaking and with proper supervision and guidence there is no task that cant be achieved.

Appendix B

Software Used

All software was developed in MATLAB. A complete recognition system was made that output the hierarchy as the system moves from frame to frame. We are usign some off-the-shelf libraries. A bvhParser is used to aid with the parsing of the bvh motion capture data files.

Appendix C

Project Schedule

The project schedules are provided in Figures C.1 and C.2. As can be seen, the project schedule was well followed in the beginning until annotation of data. This is where, due to lack of well documented help sources, the time taken to annotate data grew quite much. As a consequence, the following items were pushed forward as well. The two Gantt charts are presented next.

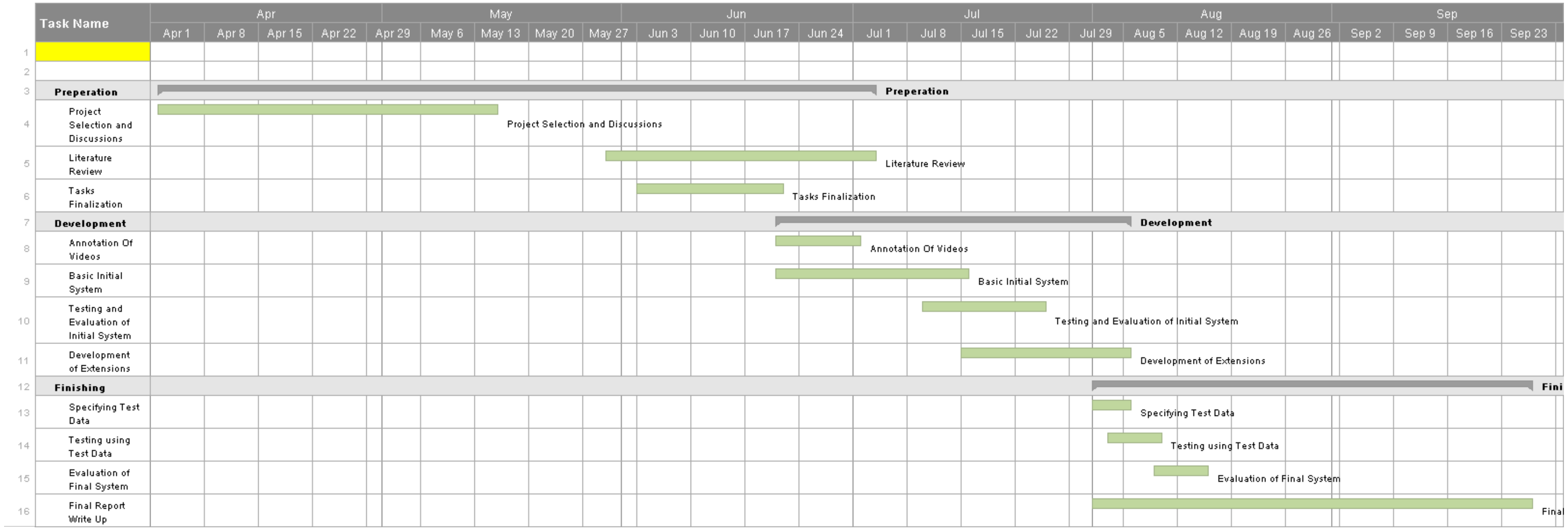


Figure C.1: Original proposed schedule for the project.

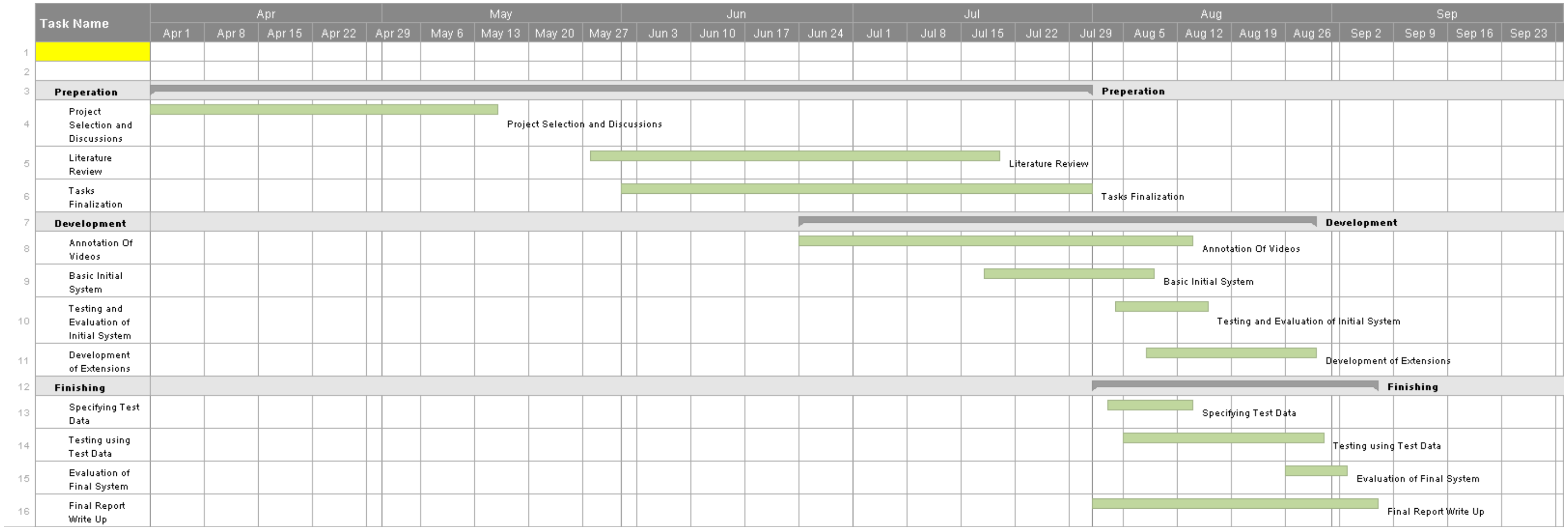


Figure C.2: Actual schedule of the project