ABSTRACT

Title of thesis: Applications of Factorization Theorem and Ontologies for Activity Modeling Recognition and Anomaly Detection

Umut Akdemir, Master of Science, 2005

Thesis directed by: Professor Rama Chellappa
Department of Electrical and Computer Engineering
Affiliate Professor in Department of Computer Science

In this thesis two approaches for activity modeling and suspicious activity detection are examined. First is application of factorization theorem extension for deformable models in two different contexts. First is human activity detection from joint position information, and second is suspicious activity detection for tarmac security. It is shown that the first basis vector from factorization theorem is good enough to differentiate activities for human data and to distinguish suspicious activities for tarmac security data.

Second approach differentiates individual components of those activities using semantic methodology. Although currently mainly used for improving search and information retrieval, we show that ontologies are applicable to video surveillance. We evaluate the domain ontologies from Challenge Project on Video Event Taxonomy sponsored by ARDA from the perspective of general ontology design principles. We also focused on the effect of the domain on the granularity of the ontology for suspicious activity detection.

Applications of Factorization Theorem and Ontologies for Activity Modeling, Recognition and Anomaly Detection

by

Umut Akdemir

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2005

Advisory Committee:

Professor Rama Chellappa, Chair/Advisor
Professor Larry Davis
Professor Ramani Duraiswami

ACKNOWLEDGEMENTS

I owe my gratitude to Professor Rama Chellappa for his invaluable lead throughout my graduate experience. He has always been supporting me with his suggestions and guidance not only in my professional research life and publications, but also in terms of the conflicts a foreigner graduate student far from home goes through. He definitely is the most considerate professor I have ever met in my life. It was a great pleasure for me to decide on what I would like to do with my professional life under his supervision. I would like to use this opportunity to express how grateful I am for his trust and empathy.

I also would like to thank Monique Thonnat and her team for providing the bank scenario videos I have worked on.

TABLE OF CONTENTS

List of Figures iv

1 Event Representation Using 3D Deformable Shape Models 1
1.1 Introduction . . . 1
1.2 Related Work . . . 3
1.3 Shape Based Activity Models . . . 7
1.3.1 Motivation . . . 7
1.3.2 Estimation of Deformable Shape Models . . . 9
1.4 Estimating Deformability of a Shape Sequence . . . 11
1.4.1 An Intuitive Understanding . . . 11
1.4.2 Computation of Deformability Index . . . 13
1.4.3 Properties of the Deformability Index . . . 15
1.5 Experimental Results . . . 16
1.5.1 Estimation of Deformability Index . . . 17
1.5.2 Shape Models for Individual Activities . . . 20
1.5.3 Shape Models for Group Activities . . . 22
1.5.4 Video Summarization in Shape Space . . . 25
1.6 Discussion: Are 3D Models Required? . . . 27

2 ONTOLOGY DRIVEN ACTIVITY MODELLING AND RECOGNITION 30
2.1 Introduction . . . 30
2.1.1 What is Ontology? . . . 30
2.1.2 What can ontology provide? . . . 30
2.1.3 Ontology Evaluation . . . 31
2.1.4 Current usage and tools for ontology . . . 31
2.2 Systematic metadata in visual environments . . . 31
2.2.1 Image Search . . . 31
2.2.2 Video Search . . . 32
2.2.3 Video Markup . . . 32
2.2.4 Video Surveillance . . . 33
2.3 Ontology in video surveillance systems . . . 34
2.3.1 Clarity in Temporal Relations . . . 35
2.3.2 Clarity of Negation Concept in Time . . . 37
2.3.3 Minimal Ontological Commitment . . . 38
2.3.4 Unified Representation . . . 40
2.4 Application example of ontology for two different complexity domains . . . 40
2.4.1 Overview . . . 40
2.4.2 Bank Scenario . . . 41
2.4.3 TSA Scenario . . . 42
2.4.4 Experimental Results . . . 43
2.5 Decision for ontology necessity in a given domain . . . 45
2.5.1 Event detection after ontology . . . 54

3 Conclusion and Future Work 56

Bibliography 58

LIST OF FIGURES

1.1 The framework for activity inference. We start by computing trajectories from a video sequence, then fit models to these trajectories (e.g. dynamic instants, Kendall's shape or 3D models as proposed here), and finally compute the similarity between model parameters for inferring about the activity. 2

1.2 Two examples of activities: (a) the binary silhouette of a walking person, and (b) people disembarking from an airplane. It is clear that both of these activites can be represented by a deformable shape model. 9

1.3 (a): Examples of the simulated shape sequence. (b): Plot of the eigenvalues, in decreasing order of magnitude, for a typical walking sequence in the USF database. 18

1.4 Plots of the first basis shape, S_1 for walk, sit and broom sequences, (a)-(c), and for jog, blind walk and crawl sequences, (d) - (f). 21

1.5 (a): The various angles used for computing the similarity of two models is shown in the Figure. The text below describes the seven dimensional vector computed from each model and whose correlation determines the similarity scores. (b): The similarity matrix for the various activities, including ones with different viewing directions and multiple cameras. The numbers correspond to the numbers in Table 1.1 for 1-16. 17 and 18 correspond to sitting and walking, where the training and test data are from two different viewing directions. 21

1.6 (a) and (b) plot the basis shapes for jogging and brooming when the viewing direction is different from the canonical one. (c) and (d) plot the rotated basis shapes. 22

1.7 (a): An example of an abnormal activity where the average trajectory is distorted to simulate an abnormal behavior. (b): Projections of the abnormal activity and a normal one on the rotated basis shapes for the first activity. 25

1.8 (a) Plot of the centered shapes formed from the average trajectories of the two activities. (b) Plot of the projections of the various instances of the two activities, as available in the training data, onto the rotated basis shapes. 26

1.9 Projections of the two activities on the rotated basis shapes for the first one are shown in (a), while the projections on the rotated basis shapes for the second one are shown in (b). 26

1.10 (a): ROC plots for classification of the two normal activities and the abnormal one. (b): A video summarization example: projections of all the motion trajectories in a three minute segment of the video sequence onto the basis shapes. The red cluster contains the projections of the passengers, the blue of the luggage cart and magenta of the airport personnel whose motion was not modeled as part of the training examples. 27

1.11 Plot of the similarity matrix for activity classification using (a) AR models, (b) ARMA model. 28

2.1 Bank attack scenario 1: Robber directly goes to counter zone, takes the employee with him and enters safe zone. After collecting valuables inside the safe he leaves the building. 46

2.2 Bank attack scenario 2: Robber directly goes to counter zone, takes the employee with him and enters safe zone. After collecting valuables inside the safe he leaves the building. This is exactly the same with attack scenario 1. Only difference is that there is a customer who runs away as soon as they enter the safe. 47

2.3 Bank attack scenario 3: 2 robbers enter the building. One of them takes the employee out of counter zone and directly goes to the vault. The other one stays inside the building to watch. After collecting valuables inside the safe they both leaves the building. Although now there are two robbers, still the robbery event itself is realized by the robber who entered safe. If he was not there it would not be counted as a robbery. 48

2.4 Bank attack scenario 4: Two robbers and a customer. One robber waits inside the building for watching and keeping the customer and the counter clerk inside the building. The other one goes to management office, takes out the manager, and uses his access to enter the safe. Still detection of unauthorized access to safe is enough to judge that this is a robbery 49

2.5 Bank no-attack scenario 1: One customer looks around to the brochures etc, while the other one is having his job done by the clerk 50

2.6 Bank no-attack scenario 2: Another no attack scenario. Very similar to the previous one. In both of these scenarios, there is no unauthorized access to safe 51

2.7 TSA scenario passengers getting on the plane: The expected procedure is exactly followed, passengers come through entrance area, they approach the plane zone and they get on. 52

2.8 TSA scenario passengers get off: Regular procedure of passengers getting off the plane. Some of them take their luggages outside, yet it is not a basic component of getting off activity as people can simply get off even when they don't have luggage. Yet clearly the overall procedure can simply be represented by ontological relations shown here. 53

Chapter 1

Event Representation Using 3D Deformable Shape Models

1.1 Introduction

Activity modeling and recognition from video sequences has applications in video surveillance and monitoring, human computer interaction, video transmission and analysis, medicine, computer graphics and virtual reality. In order to recognize different activities, it is necessary to construct an ontology of various events. Deviations from a pre-constructed dictionary can then be classified as abnormal events. It is also necessary that the representation be invariant to the viewing direction of the camera, and independent of the number of cameras (i.e. should be scalable to a video sensor network). Trajectories, usually computed from 2D video data, are a natural starting point for activity recognition systems. Trajectories contain a lot of information about the underlying event that they represent. However, one must do more than track a set of points over a sequence of images, and infer about the event from the set of tracks. Trajectories are ambiguous (different events can have the same trajectory) and depend on the viewing direction. Also, identifying events from trajectories requires the enunciation of a set of heuristics, which can vary from one instance to another of the same event. Hence, it is important to have a *proper* intermediate step in the leap from trajectories to event models (see Figure 1.1). In a recent paper, Rao, Yilmaz and Shah [65] proposed a method of representing a trajectory in terms of dramatic changes in its speed and direction. They represented a human activity in terms of action units called dynamic instants and intervals, and their method was motivated by studies on human perception. In [84], the authors proposed a shape model (along the lines of Kendall's shape theory) on the set of points in each image frame and described an activity by the dynamics of the shape.

In this paper, we propose a different approach to transition from the set of trajectories with the class of activity models. The intermediate processing step of Figure 1.1 is a 3D non-

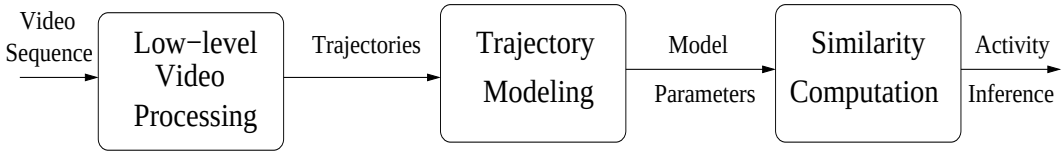


Figure 1.1: The framework for activity inference. We start by computing trajectories from a video sequence, then fit models to these trajectories (e.g. dynamic instants, Kendall's shape or 3D models as proposed here), and finally compute the similarity between model parameters for inferring about the activity.

rigid representation of the activity. The underlying hypothesis in our approach is that activities can be represented by deformable shape models, which we term as "3D event models". The 3D representation captures the 3D configuration and dynamics of the set of points taking part in the activity and is independent of the viewing direction of the camera. Also, the method works whether we have a single camera or a network of cameras looking at the scene. The 3D shape estimation is done using the factorization theorem, modified for non-rigid shapes [79, 80]. In order to properly estimate these event models, it is important to characterize the degree of non-rigidity, as this will be different for different activities. Towards this end, we propose a method to estimate the amount of deformation in a shape sequence, which we term as the "deformability index". It is obtained using spectral estimation techniques. Activities are recognized using distances between 3D models obtained from training and test sequences.

Three different kinds of experimental results are presented in order to test the efficacy of our approach. The first set of experiments is done for various activities being carried out by a single individual. We show that we are able to recognize each of those activities. We also demonstrate the view-invariant and multi-camera features of our method. In the second experiment, a group of people get off an airplane and are walking towards the terminal. We model this event and detect any abnormalities that may occur. Finally, we show the application of this approach to the problem of video summarization. Standard existing methods are used to perform low level tasks like feature detection and tracking.

In the next section, we review existing work in human activity recognition. Section 1.3 provides a justification for our shape based activity modeling framework and describes the deformable

shape estimation and activity classification methodologies. Section 1.4 presents the method for estimating the deformability index for a shape sequence and its application to the problem of computation of 3D event models. Detailed experiments are presented in Section 1.5. We would like to point out that our method for representation of 3D deformanle shapes and computation of deformability index is not specific to the event recognition problem.

1.2 Related Work

Event analysis from video sequences has a long history in the computer vision literature. We provide a brief review of past work dealing with the general problem of event recognition as well as the special situation of human activity analysis and inference.

Most of the early work on activity representation comes from the field of Artificial Intelligence (AI). One of the earliest attempts at developing a general scheme for representing activities and building a system based on it was reported by Tsuji, Morizono and Kuroda [83]. They applied their principles to understanding the activities taking place in simple cartoon films. Neumann and Novak [53] proposed a hierarchical representation of event models, with each model being a template that can be matched with scene data. Natural language descriptions of activities can be mapped on this hierarchical model. More recent work comes from the fields of image understanding and visual surveillance. The formalisms that have been employed include hidden Markov models (HMMs), logic programming and stochastic grammars. Nagel proposed [51] an early approach to obtaining conceptual descriptions from image sequences, which could then be used for representing and recognizing activities. Dousson, Gabarit and Ghallab [23], Kuniyoshi and Inoue [43], and Buxton and Gong [13] have presented models and algorithms for situation analysis from video data. Davis and Bobick [22] have developed a scheme for characterizing human actions based on the concept of "temporal templates". Bremond and Thonnat [12] have investigated the use of contextual information in activity recognition. The use of declarative models for activity recognition from video sequences was described in [68]. Each activity was represented by a set of conditions between different objects in the scene. This translated into a constraint satisfaction problem in

order to recognize the activity. A method for representing a scenario by a set of sub-scenarios and constraints combining these sub-scenarios was proposed in [86]. Castel, Chaudron and Tessier [15] developed a system for high-level interpretation of image sequences, in which they clearly separated the numerical and symbolic levels of representation and reasoning. More recently, HMMs [74, 87] have been used for recognizing American Sign Language and parametric gestures respectively. In the domain of outdoor applications, a tracking and monitoring system using a "forest of sensors" distributed around the site of interest was proposed in [28].

One of the requirements of any reliable recognition scheme is the ability to handle uncertainty. Many uncertainty-reasoning models have been actively pursued in the AI and image understanding literature, including belief networks [56] and Dempster-Shafer theory [69]. A method for inferring activities of humans and vehicles in airborne video using dynamic Bayesian networks was proposed in [7]. Large belief Networks (BNs) have been used in several video interpretation applications. For example, Intille and Bobick [36] have used large BNs to classify football plays. A system for classifying human motion and simple human interactions using small BNs was developed by Remagnino et al. [67]. A method of generating high-level descriptions of traffic scenes was implemented by Huang et al. using a dynamic Bayes' network (DBN) [35]. In [34], the authors proposed a method for recognizing events involving multiple objects using Bayesian inference. Kendall's shape theory was used to model the interactions of a group of people and objects in [84].

A specific area of research within the broad domain of event recognition is human motion modeling and analysis. The ability to recognize and track human activity using vision is one of the key challenges that must be overcome before a machine is able to interact meaningfully with a human inhabited environment. Traditionally, there has been a keen interest in studying human motion in various disciplines. In psychology, Johansson conducted classic experiments by attaching light displays to various body parts and showed that humans can identify motion when presented with only a small set of these moving dots [38]. Muybridge captured the first photographic recordings of humans and animals in motion in his famous publication on animal

locomotion towards the end of the 19-th century[50]. In kinesology the goal has been to develop models of the human body that explain how it functions mechanically [33]. The challenge to the computer vision community is to devise efficient methods to automatically track moving humans in a video sequence, reconstruct non-rigid 3D models and infer about the various activities being performed by the subjects. A survey of some of the earlier methods used in vision for tracking human movement can be found in [25]. In a more recent work, an activity recognition algorithm using dynamic instants was proposed in [65]. Kinematic chain models for human representing human motion was proposed in [11]. In [55], each human action was represented by a set of 3D curves which are quasi-invariant to the viewing direction.

The various methods listed above can be classified as either 2D or 3D approaches. 2D approaches are effective for applications where precise pose recovery is not needed or possible due to low image resolution (e.g. tracking pedestrians in a surveillance setting). However, it is unlikely that they will perform well in applications which require a high level of discrimination between various unconstrained and complex human movements (e.g. humans making gestures while walking, social interactions, dancing etc.) In such applications, 3D approaches are preferred because they can recover body pose which allows a better prediction and handling of occlusion and collision. In this paper, we estimate explicit 3D models in order to recognize various activities of an individual, as well as a group of them.

The early work on the analysis of human movement bypasses the pose recovery step altogether and uses simple, low-level 2D features from a region of interest. Models for human action are then described in statistical terms derived from these low-level features, or by simple heuristics [61, 24, 21, 62]. Another line of research involves statistical shape models (called "Active Shape Models") to determine contours [18]. A reduced parameter space of example shapes is derived using principal component analysis on feature locations used to describe those shapes. Baumberg and Hogg [8] applied Active Shape Models to track pedestrians. Motion based segmentation and tracking techniques have also been used for applications like people tracking [71]. Another class of algorithms uses explicit a priori knowledge of how the human body appears in 2D, taking essen-

tially a model-and-view based approach. These include curve-fitting with 2D ellipsoids, obtaining stick figure models [30] and orderly recognition of different body parts [5]. The problem of occlusion was considered in [44] which tracks the limbs of a silhouette by tracking anti-parallel lines. A real-time person finder system, "Pfinder" [88], was developed at MIT that models and tracks a human body using a set of "blobs", each blob described in statistical terms by spatial (x, y) and color (Y,U,V) Gaussian distribution over the pixels it consists of. Cai and Aggarwal [14] describe a system with a simplified head-trunk model to track humans across multiple cameras. Kahn and Swain [39] describe a system which uses multiple cues (intensity, edge, depth, motion) to detect people pointing laterally. More recently, Vaswani, Roy Chowdhury and Chellappa proposed a method for describing an activity of a group of people by the dynamics of the shape (defined in Kendall's sense) described on the set of moving points at each instant of time.

The general problem of 3D non-rigid shape and motion recovery from 2D images is quite difficult. However, one can take advantage of the kinematic and shape properties of a human body to make the problem tractable. One of the approaches to this problem is to solve the pose recovery problem for sub-parts and verify whether they satisfy the necessary constraints [70]. Pentland and Horowitz derived estimates of velocity and shape of a non-rigid objects from optical flow data and constraints on what kind of non-rigid motions can occur [59]. Metaxas and Terzopoulos developed a physics based framework for 3D shape and non-rigid motion estimation using dynamic models to incorporate the mechanical properties of rigid and non-rigid bodies into conventional geometric primitives [45]. The method was extended to incorporate multi-camera tracking in [40]. Another well known technique is to update pose estimates using inverse kinematics (from robot control theory) which involves inverting the mapping from the state space to the image space to obtain changes in state parameters which minimize the residual between projected model and image features [66]. Gavrila and Davis follow a different approach in which the measurement equation is directly used to synthesize a model and a fitting measure between synthesized and observed features is used for feedback [26]. Azarbayejani and Pentland recover 3D shape and orientation from 2D "blob" features using non-linear estimation techniques [58]. In another work [57], Pentland

used deformable superquadrics to fit range data. Most explicit model based approaches assume certain domain constraints like calibrated cameras, known background, initial pose and uncluttered environments. In contrast, the methods which work purely from image data are very specialized to given training data. Bregler proposed a combination of layered image representations with dynamic models and Hidden Markov Models in a probabilistic framework in order to find the right balance between structure estimation and learned parameters [10]. In [11], the authors used the twist and product of exponential map formalism for kinematic chains [49] for modeling the motion of different body parts attached by body joints. Ioffe and Forsyth introduce a "mixture of trees" formalism to track a human body by identifying candidate primitives and then grouping them so as to satisfy the constraints on the relative configuration of the parts [37]. A flow based tracking scheme was introduced in [81, 80] which approximates a non-rigid object by a composition of known shapes, thus limiting the rank of the measurement matrix (of the entire image sequence). Mori and Malik [48] have recently proposed an algorithm to estimate body configuration and pose in 3D space from a single image by matching a test shape against a database which stores a number of exemplar 2D views of a human body by shape context matching and kinematic chain based deformation model. An exemplar-based, probabilistic paradigm for visual tracking without estimating 3D pose was proposed in [82]. A probabilistic 3D tracking algorithm using shape-encoded particle propagation was presented in [47]. In [55], each human action was represented by a set of 3D curves which are invariant to the viewing direction.

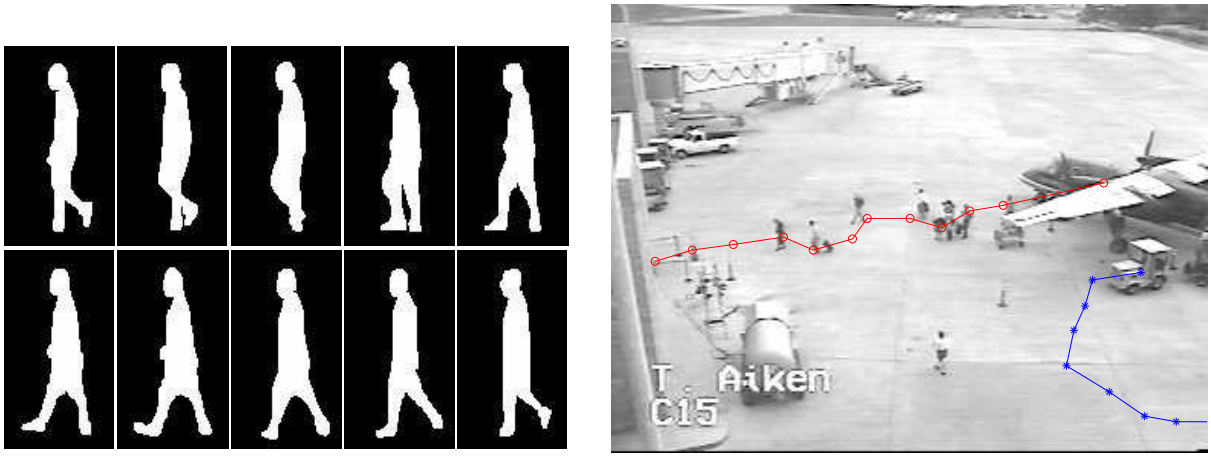
1.3 Shape Based Activity Models

1.3.1 Motivation

In this paper, we propose a framework for recognizing activities by first computing the trajectories of the various points taking part in the activity, followed by a non-rigid 3D shape model, estimated from the trajectories. It is based on the empirical observation that many activities have an associated structure and a dynamical model. Consider, as an example, the set of images of a walking person in Figure 1.2(a) (obtained from the USF database for the Gait Challenge problem

[60]). The binary representation is used to clearly show the change in shape of the body for one complete walk cycle. The person in this figure is free to move his/her hands and feet any way he/she likes. However, this random movement does not constitute the activity of walking. For humans to perceive and appreciate the walk, the different parts of the body have to move in a certain synchronized manner. In mathematical terms, this is equivalent to modeling the walk by the deformations in the shape of the body of the person. Similar comments can be made for other activities performed by a single human, e.g. dancing, jogging, sitting, etc. An analogous example can be provided for an activity involving a group of people. Consider people getting off a plane and walking to the terminal, where there is no jet-bridge to constrain the path of the passengers (Figure 1.2(b)). Every person after disembarking, is free to move as he/she likes. However, this does not constitute the activity of people getting off a plane and heading to the terminal. The activity here is comprised of people walking along a path that leads to the terminal. Again, we see that the activity can be modeled by the shape of the trajectories taken by the passengers. Using deformable shape models is a higher level abstraction of the individual trajectories and provides a method of analyzing all the points of interest together, thus modeling their interactions in a very elegant way.

Not only is the activity represented by a deformable shape sequence, the amount of deformation is different for different activities. For example, it is reasonable to say that the shape of the human body while dancing is usually more deformable than that when walking, which is more deformable than when standing still. Since it is possible for the human observer to obtain an idea of deformability based on the contents of the video sequence, the information about how deformable a shape is must be contained in the sequence itself. We will use this intuitive notion to quantify the deformability of a shape sequence from a set of tracked points on the object. In our activity represententation model, a deformable shape is represented as a linear combination of rigid basis shapes. The deformability index will provide a theoretical method for estimating the number of basis shapes required.



(a) (b)

Figure 1.2: Two examples of activities: (a) the binary silhouette of a walking person, and (b) people disembarking from an airplane. It is clear that both of these activites can be represented by a deformable shape model.

1.3.2 Estimation of Deformable Shape Models

We hypothesize that each shape sequence can be represented by a linear combination of 3D basis shapes. Mathematically, if we consider the trajectories of P points representing the shape (e.g. landmark points), then the overall configuration of the P points is represented as a linear combination of the basis shapes as

$$S = \sum_{i=1}^{K} l_i S_i, \qquad S, S_i \in \Re^{3\times P}, l \in \Re. \tag{1.1}$$

The choice of K is determined by quantifying the deformability of the shape sequence and will be the studied in detail in Section 1.4. We will assume a weak perspective projection model for the camera.

A number of methods exist in the computer vision literature for estimating the basis shapes. In [79], the authors considered P points tracked across F frames in order to obtain two $F \times P$ matrices $\mathbf{U}$ and $\mathbf{V}$. Each row of $\mathbf{U}$ contains the x-displacements of all the P points for a specific time frame, and each row of $\mathbf{V}$ contains the corresponding y-displacements. It was shown in [79], that for 3D rigid motion under orthographic camera model, the rank, r, of $\begin{bmatrix}\mathbf{U}\\ \mathbf{V}\end{bmatrix}$ has an upper

bound of 3. The rank constraint is derived from the fact that $\left[\begin{array}{c}\mathbf{U}\\ \hline \mathbf{V}\end{array}\right]$ can be factored into two matrices $\mathbf{M}_{2F\times r}$ and $\mathbf{S}_{r\times P}$, corresponding to the pose and 3D structure of the scene, respectively. In [80], it was shown that for non-rigid motion, the above method could be extended to obtain a similar rank constraint, but one that is higher than the bound for the rigid case. We will adopt the last mentioned method for computing the basis shapes. We will outline the basic steps of their approach in order to clarify the notation for the remainder of the paper.

Given F frames of a video sequence with P moving points, we can obtain the trajectories of all these points over the entire video sequence. These P points can be represented in a measurement matrix as

$$\mathbf{W}_{2F\times P} = \begin{bmatrix} u_{1,1} & \cdots & u_{1,P} \\ v_{1,1} & \cdots & v_{1,P} \\ \vdots & \vdots & \vdots \\ u_{F,1} & \cdots & u_{F,P} \\ v_{F,1} & \cdots & v_{F,P} \end{bmatrix}, \tag{1.2}$$

where $u_{f,p}$ represents the x-position of the p^{th} point in the f^{th} frame and $v_{m,p}$ represents the y-position of the same point. Under weak perspective projection, the P points of a configuration in a frame f, are projected onto 2D image points $(u_{f,i}, v_{f,i})$ as

$$\begin{bmatrix} u_{f,1} & \cdots & u_{f,P} \\ v_{f,1} & \cdots & v_{f,P} \end{bmatrix} = \mathbf{R}_f \left(\sum_{i=1}^{K} l_{f,i} S_i \right) + \mathbf{T}_f, \tag{1.3}$$

where,

$$\mathbf{R}_f = \begin{bmatrix} r_{f1} & r_{f2} & r_{f3} \\ r_{f4} & r_{f5} & r_{f6} \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{R}_f^{(1)} \\ \mathbf{R}_f^{(2)} \end{bmatrix}. \tag{1.4}$$

$\mathbf{R}_f$ represents the first two rows of the full 3D camera rotation matrix and $\mathbf{T}_f$ is the camera translation. The translation component can be eliminated by subtracting out the mean of all the 2D points, as in [79]. We now form the measurement matrix $\mathbf{W}$, which was represented in (1.2), with the means of each of the rows subtracted. The weak perspective scaling factor is implicitly

coded in the configuration weights, $\{l_{f,i}\}$.

Using (1.2) and (1.3), it is easy to show that

$$\mathbf{W} = \begin{bmatrix} l_{1,1}\mathbf{R}_1 & \cdots & l_{1,K}\mathbf{R}_1 \\ l_{2,1}\mathbf{R}_2 & \cdots & l_{2,K}\mathbf{R}_2 \\ \vdots & \vdots & \vdots \\ l_{F,1}\mathbf{R}_F & \cdots & l_{F,K}\mathbf{R}_F \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_K \end{bmatrix} \tag{1.5}$$

$$= \mathbf{Q}_{2F\times 3K}.\mathbf{B}_{3K\times P}, \tag{1.6}$$

which is of rank $3K$. The matrix $\mathbf{Q}$ contains the pose for each frame of the video sequence and the weights $l_1, ..., l_K$. The matrix $\mathbf{B}$ contains the basis shapes corresponding to each of the activities. In [80], it was shown that $\mathbf{Q}$ and $\mathbf{B}$ can be obtained using singular value decomposition (SVD), and retaining the top $3K$ singular values, as $\mathbf{W}_{2M\times P} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ and $\mathbf{Q} = \mathbf{U}\mathbf{D}^{\frac{1}{2}}$ and $\mathbf{B} = \mathbf{D}^{\frac{1}{2}}\mathbf{V}^T$.

1.4 Estimating Deformability of a Shape Sequence

The above mentioned rank constraint requires knowledge of K in order to estimate the shape and motion parameters. This is usually determined heuristically from the physics of the object whose structure is being estimated. We now provide a theoretical method for estimating K by reinterpreting the above equations in stochastic framework. Our method is non-iterative, does not require determination of a threshold (as in other methods based on minimum error thresholding [41]) and does not need an initial guess, which is usually based on heuristics. In turn, it leads to a definition of deformability of a shape sequence.

1.4.1 An Intuitive Understanding

In modeling the dynamics of shape evolution, it is important to separate out the "global" motion of the shape (i.e., the translation and rotation) from its "deformation", an issue that was analyzed in [73]. While there are well-defined measures for the global motion of an object, quantitative measures of its deformations are less well known. We address this issue of quantifying the de-

formation of a shape sequence by defining a "deformability index". For a rigid shape (i.e., the shape does not change from one image frame to the next), the deformability index is one. We show how to derive this index in shape space using tools from spectral analysis. Experiments on real-life data of human activities are carried out (see Section 1.5) and the results are in accordance with our intuitive judgment of the deformation involved in those activities and corroborate certain experimental findings in human gait analysis.

As a shape deforms, the position of the set of points defining the shape changes from one image frame to the next. The change in the position of this sequence of points determines how much the shape is changing, e.g. whether it is being squeezed or expanded or remaining the same. Defining a deformability index depends on the ability to obtain a mathematical description of this shape change. As explained before, we model a shape sequence that deforms over time as a composition of a number of basis shapes, where the weight given to each basis shape changes with time, thus leading to deformations in the original shape. It is usually the case that more deformable a shape is, more is the number of basis shapes required to represent it. However, there is no well-defined criterion for estimating the number of basis shapes. At a minimum, a rigid shape would require only one basis shape, while there is no theoretical upper limit. Therefore we need a method to estimate the number of basis shapes from the point sequence.

The theoretical derivation which follows does precisely this. It proceeds by using the transformation of the point sequence to a shape space (as shown in Section 1.3) and estimating the dimensionality of this shape space. Spectral analysis provides a method for achieving this purpose. The dimensionality of the shape space will determine the deformability index. The noise in the sequence of feature positions will be taken into account in order to correctly estimate the deformability index. Since the noise can randomly alter the positions of the points, it can give a false notion of increased variability in the shape sequence, leading to a higher dimensionality of the shape space. Also, rigid 3D transformations of the shape can provide the impression of deformation. This will be factored out in estimating the deformability index. However, estimation of 3D structure will not be required for this purpose.

1.4.2 Computation of Deformability Index

Consider the set of coordinates representing the shape of the deformable object in a particular frame of a video sequence to be the realization of a random process. The sequence of frames depicts the deformation of the shape, along with the effects of the 3D translation and rotation. Represent the x and y coordinates of the sampled points in a single frame as a vector $\mathbf{y} = [u_1, ..., u_P, v_1, ..., v_P]^T$. Then, from (1.6), it is easy to show that for K basis shapes (K is unknown)

$$\mathbf{y}^T = \left[l_1\mathbf{R}^{(1)}, ..., l_K\mathbf{R}^{(1)}, l_1\mathbf{R}^{(2)}, ..., l_K\mathbf{R}^{(2)}\right] * \begin{bmatrix} S_1 & \\ \vdots & 0 \\ S_k & \\ & S_1 \\ 0 & \vdots \\ & S_k \end{bmatrix} + \eta^T, \tag{1.7}$$

$$i.e., \quad \mathbf{y} = (\mathbf{q}_{1\times 6K}\mathbf{b}_{6K\times 2P})^T + \eta$$

$$= \mathbf{b}^T\mathbf{q}^T + \eta, \tag{1.8}$$

where η represents the noise in the sequence of tracked points and is assumed to be a zero-mean random process. The vector $\mathbf{q}$ is obtained by juxtaposing two consecutive rows of $\mathbf{Q}$, corresponding to the same image frame, in equation (1.6). The matrix $\mathbf{b}$, which is constant across all the frames, is obtained by duplicating $\mathbf{B}$ in equation (1.6), as shown in equation (7).

Assuming that the coordinates of the points representing the shape in all the F frames can be considered to be realizations of the same random process (which is a reasonable assumption since they represent the same shape), with possibly different noise statistics, we can compute the correlation matrix of $\mathbf{y}$. Let $\mathbf{R_y} = E[\mathbf{y}\mathbf{y}^T]$ be the correlation matrix of $\mathbf{y}$ and $\mathbf{C}_\eta$ the covariance matrix of η. Hence,

$$\mathbf{R_y} = \mathbf{b}^T E[\mathbf{q}^T\mathbf{q}]\mathbf{b} + \mathbf{C}_\eta. \tag{1.9}$$

The correlation matrix, $\mathbf{R_y}$, is of size $2P \times 2P$ and can be estimated from the sequence of points

representing the shapes as $\mathbf{R_y} = \frac{1}{F}\sum_{f=1}^{F} \mathbf{y}_f \mathbf{y}_f^T$, where $\mathbf{y}_f$ is the vector $\mathbf{y}$ (defined above) in the frame f. The expectation on the right hand side of equation (1.9) can be computed similarly as $E[\mathbf{q}^T\mathbf{q}] = \frac{1}{F}\sum_{f=1}^{F} \mathbf{q}_f^T\mathbf{q}_f$, where $\mathbf{q}_f$ is the vector $\mathbf{q}$ (defined above) for frame f and is obtained from the matrix $\mathbf{Q}$ in equation (1.6).

The noise covariance matrix, C_η, represents the accuracy with which the feature points are tracked and needs to be estimated from the image frames. Since η need not be an independent and identically distributed (IID) noise process, C_η will not necessarily have a diagonal structure (but it is symmetric and positive semi-definite). For the purposes of setting a precise threshold (which will become clear soon), it is desirable that $\mathbf{C}_\eta$ be a diagonal matrix.

Consider the diagonalization of $\mathbf{C}_\eta = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^T$, where $\mathbf{\Lambda} = \text{diag}[\mathbf{\Lambda}_s, 0]$ and $\mathbf{\Lambda}_s$ is an $L \times L$ matrix of non-zero singular values of $\mathbf{\Lambda}$. Let $\mathbf{P}_s$ denote the orthonormal columns of $\mathbf{P}$ corresponding to the non-zero singular values. Therefore,

$$\mathbf{C}_\eta = \mathbf{P}_s\mathbf{\Lambda}_s\mathbf{P}_s^T. \tag{1.10}$$

Premultiplying equation (1.8) by $(\mathbf{P}_s\mathbf{\Lambda}_s^{\frac{1}{2}})^{-1}$, we see that (1.8) becomes

$$\tilde{\mathbf{y}} = \tilde{\mathbf{b}}^T\mathbf{q}^T + \tilde{\eta}, \tag{1.11}$$

where $\tilde{\mathbf{y}} = \mathbf{\Lambda}_s^{-\frac{1}{2}}\mathbf{P}_s^T\mathbf{y}$ is a $L \times 1$ vector, $\tilde{\mathbf{b}}^T = \mathbf{\Lambda}_s^{-\frac{1}{2}}\mathbf{P}_s^T\mathbf{b}^T$ is a $L \times 6K$ matrix and $\tilde{\eta} = \mathbf{\Lambda}_s^{-\frac{1}{2}}\mathbf{P}_s^T\eta$. It can be easily verified that the covariance of $\tilde{\eta}$ is an identity matrix $\mathbf{I}_{L\times L}$. This is known as the process of "whitening", whereby the noise process is transformed to be IID [75].

Representing by $\mathbf{R}_{\tilde{\mathbf{y}}}$ the correlation matrix of $\tilde{\mathbf{y}}$, it can be seen that

$$\mathbf{R}_{\tilde{\mathbf{y}}} = \tilde{\mathbf{b}}^T E[\mathbf{q}^T\mathbf{q}]\tilde{\mathbf{b}} + \mathbf{I} \;=\; \mathbf{\Delta} + \mathbf{I}, \tag{1.12}$$

where, for simplicity, $\mathbf{\Delta} \stackrel{\Delta}{=} \tilde{\mathbf{b}}^T E[\mathbf{q}^T\mathbf{q}]\tilde{\mathbf{b}}$. Now, $\mathbf{R}_{\tilde{\mathbf{y}}}$ is of dimension $L \times L$, $\tilde{\mathbf{b}}^T$ is of rank $L \times 6K$ and $E[\mathbf{q}^T\mathbf{q}]$ is of rank $6K \times 6K$. Thus, $\mathbf{\Delta}$ has maximum rank $6K$, where K is the number of

basis shapes (assuming $L > 6K$). This is based on the fact that if $\mathbf{A}_{m\times n} = \mathbf{F}_{m\times r}\mathbf{G}_{r\times n}$, then the Rank$(\mathbf{A}) \leq r$. For a general 3D scene undergoing translation and rotation, the rank will be $6K$, which is the case we will consider below. Representing by $\mu_i(\mathbf{A})$ the i^{th} eigenvalue of the matrix $\mathbf{A}$, we see that

$$\begin{aligned} \mu_i(\mathbf{R}_{\tilde{\mathbf{y}}}) = &\quad \mu_i(\mathbf{\Delta}) + 1, \quad \text{for} \quad i = 1, ..., 6K, \quad \text{and} \\ \mu_i(\mathbf{R}_{\tilde{\mathbf{y}}}) = &\quad 1, \qquad\quad \text{for} \quad i = 6K+1, ..., L. \end{aligned} \tag{1.13}$$

Hence, there are $6K$ eigenvalues above 1. By counting the number of eigenvalues that are greater than 1 and dividing it by 6, we can obtain an estimate of K, which is the dimensionality of the shape space represented by the sequence of deforming points. Since K denotes the number of basis shapes that can model the feature point sequence, it provides a measure of the deformability of the shape sequence. The more the number of basis shapes required to model a shape sequence, the more deformable it is. Thus, for a general 3D scene undergoing translation and rotation, we have

$$\text{Deformability Index} = \frac{\#\text{eigenvalues of } \mathbf{R}_{\tilde{\mathbf{y}}} > 1}{6}. \tag{1.14}$$

1.4.3 Properties of the Deformability Index

- For the case of a 3D rigid body, the deformability index is 1. In this case, the only variation in the values of the vector $\mathbf{y}$ from one image frame to the next is due to the global rigid translation and rotation of the object. The rank of the matrix $\mathbf{\Delta}$ will be 6 [80, 79] and the deformability index will be 1.

- For the special case of a planar scene, the corresponding rank of $\mathbf{\Delta}$ would be $4K$, and thus the deformability index should be calculated by dividing the number of eigenvalues over 1 by 4.

- Estimation of the deformability index does not require explicit computation of the 3D structure and motion in equation (1.6), since we need to compute the eigenvalues of the covariance

matrix of the 2D feature positions. In fact, for estimating the shape and rotation matrices in equation (1.6) it is essential to know the value of K. Thus the method outlined in Section 1.4.2 should precede computation of the shape in Section 1.3. Using our method, it is possible to obtain an algorithm for deformable shape estimation without having to guess the value of K.

- The computation of the deformability index takes into account any rigid 3D translation and rotation of the object (as recoverable under a scaled orthographic camera projection model), even though it has the simplicity of working only with the covariance matrix of the 2D projections. Thus it is more general than a method that considers purely 2D image plane motion.

- The "whitening" procedure described above enables us to choose a *fixed* threshold of one for comparing the eigenvalues.

1.5 Experimental Results

We applied our method for two very different types of events. In the first, we recognize the activities performed by an individual, e.g. walking, running, sitting, crawling, etc. We show that the use of 3D models as the intermediate step allows us to recognize these activities independent of the viewing direction. We also show how activity recognition can be done in a multi-camera framework. In the second set of experiments, we show the effect of our method in recognizing the activities of a group of people, represented as point objects. We show that we can identify abnormal behavior in this group. The application of this method to the problem of video summarization and discovering new activities is also discussed.

Our activity recognition algorithms are based on the computed 3D models and consist of a learning/training phase and a testing one. During the training phase, the 3D models for various activities are computed. Given a test sequence, the 3D model estimated from this sequence is compared with that learned before and a similarity score is computed based on a measure of the difference of the two 3D models. The exact method for computing this difference is based on the

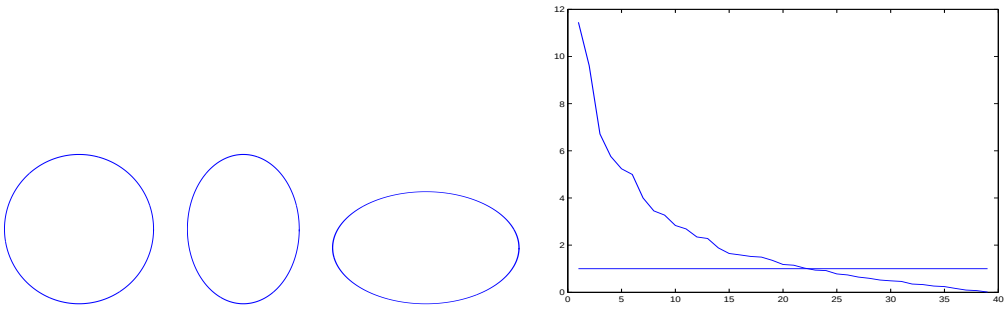
particular application and is explained in detail later. However, before it is possible to compute the 3D models, we need to estimate the number of basis shapes required to represent the deformable shape sequence. For this reason, we first present our results on the deformability index.

1.5.1 Estimation of Deformability Index

Experimental evaluation of the theory for estimation of the deformability index was carried out on simulation data and real life imagery.Next, we applied our theory to walking sequences of humans as available in the USF Gait Challenge Database [60]. Here we found that our deformability estimates are in accordance with some of the results on shape representation reported in the gait recognition literature. In both the experiments, the shapes were centered in the image frames, scaled and aligned so as to make the human body upright.

Experiments With Simulation Data The first set of experiments was conducted by simulating a deforming shape as a combination of rigid shapes. The aim was to test the validity of the theory in predicting the deformability index, when the number of basis shapes is known. We simulated a sequence of deformable shapes as a combination of two rigid shapes. Examples of the shape sequence, projected to the 2D image plane, are shown in Figure 1.3(a). The shape was sampled uniformly around its boundary starting at a fixed point, thereby maintaining correspondence between the different frames. Noise of known variance was added to the feature positions. We estimated the deformability index using the theory described in Section 1.4.2. When the noise variance was low (about 10 pixel variance), the number of basis shapes (i.e. the deformability index) was correctly estimated to be 2. As the noise was increased, the error in the estimate of the deformability index was higher, but the estimate was never more than 3 (for a standard deviation of about 7 pixels). This simple experiment serves as the initial validation of the estimate of the deformability index.

Experiments With Motion Capture Data In this experiment, we computed the deformability index of the human body for a large number of activities, and found them to be very consistent with what would be expected intuitively by a human observer. We used the motion-capture data



(a) (b)

Figure 1.3: (a): Examples of the simulated shape sequence. (b): Plot of the eigenvalues, in decreasing order of magnitude, for a typical walking sequence in the USF database.

available from Credo Interactive Inc. and Carnegie Mellon University in the BioVision Hierarchy and Acclaim formats. It has a number of examples of different activities and is thus a rich dataset for studying shape sequences. [1] The combined dataset included a number of subjects performing various activities, like walking, jogging, sitting, crawling, brooming, etc. For each of these activities, we had multiple video sequences. Also, many of the activities contained video from different viewpoints.

Using the video sequences and the theory outlined in Section 1.3, we computed the 3D basis shapes and their combination coefficients (see equation (1.1)). The first basis shapes are shown in Figure 1.4 for six different activities.

For the different activities in this database, we computed the deformability index from equation (1.14). The deformability index, computed for each of these sequences, is shown in Table 1. Since this value denotes the number of basis shapes required to represent the video sequences, we resynthesized the original sequences using the basis shapes and combination coefficients obtained from equation (1.6). Equation (1.3) was used for the synthesis and the value of K was determined by the procedure in Section 1.4.2. In all the cases, the error at none of the feature points was more than 1 pixel.

From Table 1.1, a number of interesting observations can be made. For the walk sequences, the deformability index was between 5 and 6. This matches the hypotheses in papers on gait recognition where it is mentioned that about five exemplars are necessary to represent a full cycle

[1] While there are a number of standard datasets for shapes, we could not find any large datable for the study of shape sequences.

Table 1.1: Deformability Index for Human Activities Using Motion Capture Data

	Activity	Deformability Index		Activity	Deformability Index
1	Walk (Seq. 1)	5.8	10	Broom (Seq. 2)	8.8
2	Walk (Seq. 2)	4.7	11	Jog	5.0
3	Fast Walk	8.0	12	Blind walk	8.8
4	Walk while throwing hands around	6.8	13	Crawl	8.0
5	Walk with drooping head	8.8	14	Jog while taking U-turn (Seq. 1)	4.8
6	Sit (Seq. 1)	8.0	15	Jog while taking U-turn (Seq. 1)	5.0
7	Sit (Seq. 2)	8.2	16	Broom in a circle	9.0
8	Sit (Seq. 3)	8.2	17	Female Walk	7.0
9	Broom (Seq. 1)	7.5	18	Slow Dance	8.0

of gait [41]. The number of basis shapes increases for fast walk, as expected from some of the results in [78]. When the person walks doing some other things (like moving head or hands or a blind person's walk), the number of basis shapes needed to represent it (i.e. the deformability index) increases from that of normal walk. The result that might seem surprising initially is the high deformability index for sitting sequences. On closer examination though, it was found that the person, while sitting, was making all kinds of random gestures as in talking to someone else. That increased the deformability index for these sequences. Also, the deformability index is insensitive to changes in viewpoint (azimuth angle variation only), as can be seen by comparing the jog sequences (14 and 15 with 11) and broom sequences (16 with 9 and 10). This is not surprising since we do not expect the deformation of the human body to change due to rotation about the vertical axis. The deformability index, thus calculated, is used to estimate the 3D shape, some of which are shown in Figure 1.4 and which will be used later for activity recognition experiments.

Experiments on Gait Dataset The USF Gait Challenge Dataset [60] was used for our experiments because of two reasons. It has a number of examples of different people walking under different conditions. Thus it would allow us to test the consistency of the estimates for the deformability index. Secondly, a number of researchers have reported results in this dataset and thus we would be able to corroborate our conclusions with their results.

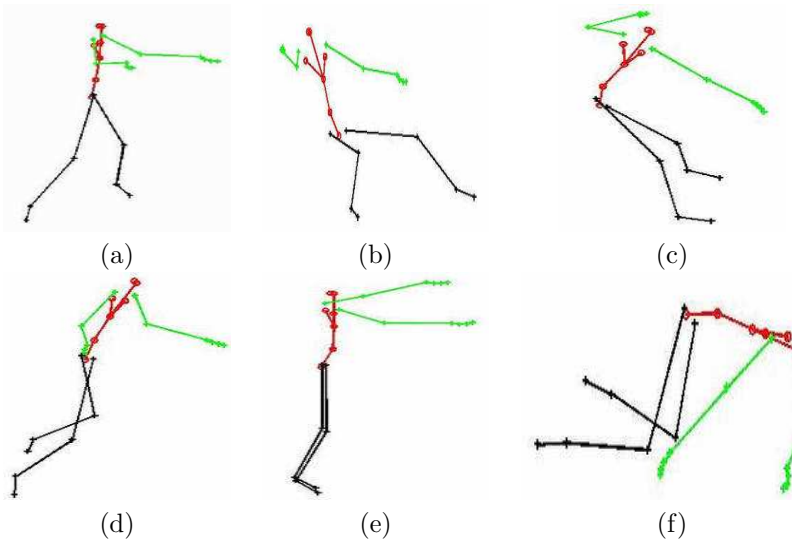
We used the background subtracted images of the walking person, when the person is presenting a side view to the camera, as shown in Figure 1.2(a). The outer boundary of the person

was sampled in order to obtain the shape vector. The method described in [76] was adopted to estimate the variance of the noise in the feature positions from the original images. The method uses the inverse of the Hessian matrix of the second-order partial derivatives of the intensity along the horizontal and vertical axes. By using the same number of sample points in each frame, an approximate correspondence was maintained between the feature points in the different frames. We experimented with 10 subjects walking on grass and concrete surfaces and wearing different types of shoes. For all the cases, the deformability index ranged from 3.8 to 5.2. Figure 1.3(b) shows a typical plot of the eigenvalues arranged in descending order of magnitude along with the threshold of one. It has been noted in [41] that four to five exemplars are needed to represent a complete cycle of gait, using a minimum error thresholding method. Our analysis provides a theoretical justification for the choice of the number of exemplars.

1.5.2 Shape Models for Individual Activities

We classify the various activities performed by an individual using the motion capture data described in the previous section. Using the video sequences and the theory outlined in Sections 1.3 and 1.4, we compute the basis shapes and their combination coefficients (see equation (1.1)). We found that the first basis shape, S_1, contained most of the information. The estimated first basis shapes are shown in Figure 1.4 for six different activities. Since the values of l_i are small for $i > 1$, we used only the first basis shape to compute the similarity between the various activities. In order to compute the similarity, we considered the various joint angles between the different parts of the estimated 3D models. The angles considered are shown in Figure 1.5(a). The idea of considering joint angles for activity modeling has been used before, e.g. in gait recognition [77]. We considered the seven dimensional vector obtained from the angles as shown in Figure 1.5(a). The correlation between two such angle vectors was used as the measure of similarity.

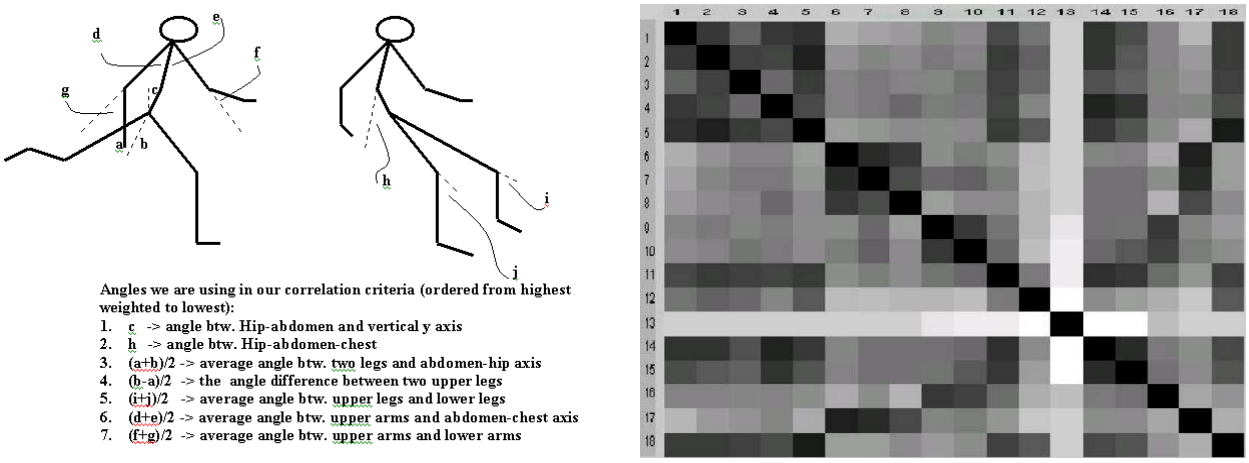
The similarity matrix is shown in Figure 1.5(b). For the moment, consider the upper 13 x 13 block of this matrix. We find that the different walk sequences are close to each other. Similarly for the sitting and brooming sequences. The jog sequence, besides being closest to itself, is also



(a) (b) (c)

(d) (e) (f)

Figure 1.4: Plots of the first basis shape, S_1 for walk, sit and broom sequences, (a)-(c), and for jog, blind walk and crawl sequences, (d) - (f).

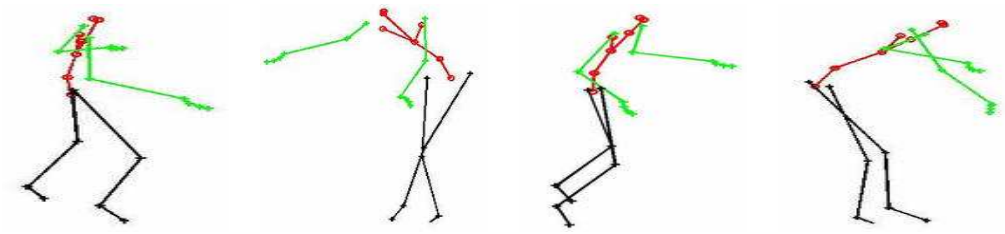


(a) (b)

Figure 1.5: (a): The various angles used for computing the similarity of two models is shown in the Figure. The text below describes the seven dimensional vector computed from each model and whose correlation determines the similarity scores. (b): The similarity matrix for the various activities, including ones with different viewing directions and multiple cameras. The numbers correspond to the numbers in Table 1.1 for 1-16. 17 and 18 correspond to sitting and walking, where the training and test data are from two different viewing directions.

close to the walk sequences. Blind walk is close to jogging and walking. The crawl sequence does not match any of the rest and this is clear from Row 13 of the matrix. Thus, the results obtained using our method are reasonably close to what we would expect from a human observer.

Next, we consider the situation where we try to recognize activities when the input video sequences are from different viewpoints. This is the most interesting part of the method, as it



(a) (b) (c) (d)

Figure 1.6: (a) and (b) plot the basis shapes for jogging and brooming when the viewing direction is different from the canonical one. (c) and (d) plot the rotated basis shapes.

demonstrates the strength of using 3D models for activity recognition. In our dataset, we had three sequences where the motion is not parallel to the image plane, two for jogging in a circle and one for brooming in a circle. We considered a portion of these sequences where the person is not parallel to the camera. From each such video sequence, we computed the basis shapes. This basis shape is rotated, based on an estimate of its pose, and transformed to the canonical plane (i.e. parallel to the image plane). The basis shapes before and after rotation are shown in Figure 1.6. This rotated basis shape is used to compute the similarity of this sequence with the others, exactly as described above. Rows 14-18 of the similarity matrix shows the recognition performance for this case. The jogging sequences are close to the jogging in the canonical plane (Column 11), followed by walking along the canonical plane (Columns 1-6). For the broom sequence, it is closest to the brooming in the canonical plane (Column 9 and 10). The sitting and walking sequences (columns 17 and 18) are close to the other sitting and walking sequences, even though they are captured from different viewing directions.

1.5.3 Shape Models for Group Activities

In this section, we consider a very different kind of activity. A group of people get off an airplane and walk to the terminal. Also, there are other moving objects like vehicles, airport personnel, etc. The goal is to classify the activities of the different groups of objects (people vs. vehicles) and to identify an abnormal behavior (e.g. a passenger straying from the normal path), using the information available in the trajectories.

Given a video sequence with each moving point representing the motion of a different object,

we can obtain the trajectories of all these points over the entire video sequence. The trajectory defines the particular activity. For the case of people getting off an airplane, each person is represented by a point. An average trajectory over all the people represents the activity of people getting of the plane. If we have M different training video sequences with different instances of the same activity, we can obtain many such example trajectories. Each of the example trajectories can be sampled uniformly to produce a set of P points, each represented as a pair of x and y co-ordinates. Note that the number of rows in the matrix $\mathbf{W}$ in (1.2) depends on the number of training sequences, i.e. $F = M$.

During **training**, we compute the rotation matrix and the average shapes as explained above. For the m^{th} video sequence, consider the rows $(2m-1)$ and $2m$ of the matrix $\mathbf{W}$, and represent it by W_m. It represents the average trajectory of the activities in the m^{th} training sequence. From (1.3), we see that $l_{m,i}$ can be computed by taking the inner product of W_m with $\mathbf{R}_m S_i$, i.e.

$$l_{m,i} =< W_m, \mathbf{R}_m S_i > \tag{1.15}$$

for each activity $i = 1, ..., N$ and for each training video sequence $m = 1, ..., M$. Thus for each activity i, we have M values of l_i. These multiple values of l_i represent a significant part of the range of values that can be taken by different instances of these activities. Since a fixed camera is looking at the same set of activities, the rotation matrices will not be very different between the different instances of the same activity. Hence, all the l_i for each activity cluster together and can be used for recognition.

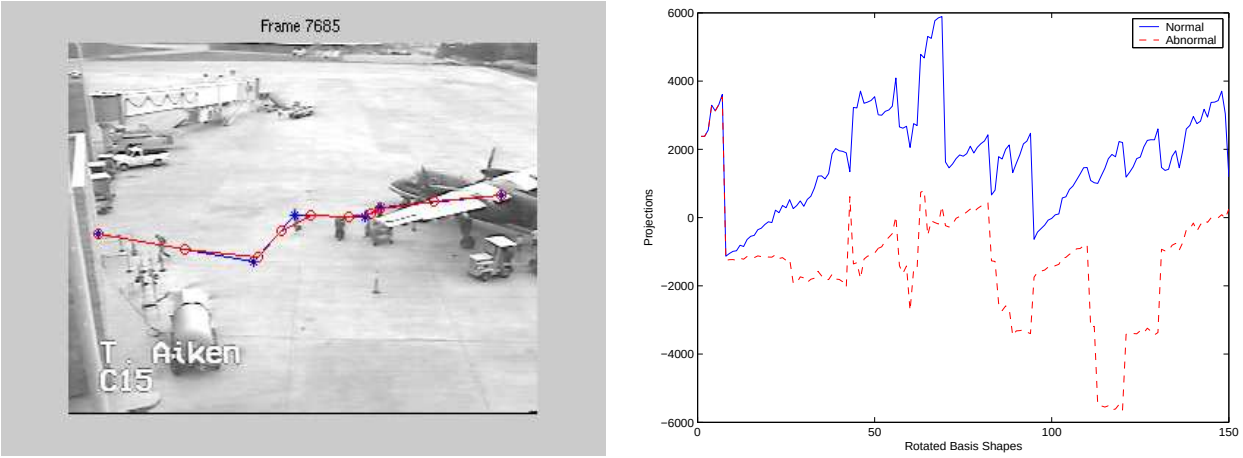
During **testing**, we consider the trajectory of each object in the video sequence. The procedure described above can be re-applied to the set of tracked points in the sequence in order to obtain the configuration weights by projecting onto the rotated basis shapes, as in (1.15). The cluster to which the computed l_i belong can be used to identify the activity. The intuitive idea is that the set of weights learned from the training examples cover most of the possible ones for normal activities. Thus, if projections for the test activity lie within a cluster for one of the activities, then we can claim to have recognized that particular activity. In practice, we can set a

threshold, $T < M$, for the number of projections that need to lie within a cluster for the activity to be recognized as such. By this method, the activity of each object is individually detected and verified in this 3D shape space. One of the advantages of our method is that it is computationally very inexpensive, since all that it does for classification and verification is to compute projections of tracked features onto basis shapes learned a-priori.

In the airport surveillance situation, the trajectories of the main objects are obtained using a motion detection and tracking algorithm. In Figure (1.8)(a), we plot the average centered shapes (i.e. after the mean of every row of $\mathbf{W}$ is subtracted out) for the two major activities, passengers disembarking and the path of the luggage cart or fuel tank. The airport personnel are identified a-priori and their motion is neglected for the purposes of this analysis. It is clear from the plot that the shapes are very different, and successfully exploiting them can lead to a good classification algorithm for the various activities. Also, when an abnormal event occurs (Figure (1.7)(a)), the trajectory, as represented by the shape, is significantly deformed and can be identified.

The plot of the various values of $l_{m,1}$ and $l_{m,2}$ for all m, learned from the training sequences, is shown in Figure 1.8(b), thus showing the clear demarcation between the two activities. In Figure (1.9)(a), we show the plots of the projections of the activity of passengers deplaning on the two sets of rotated basis shapes, learned during the training phase, i.e. $\mathbf{R}_m S_1$ and $\mathbf{R}_m S_2$, for $m = 1, ..., 150$. Another test case is the motion of the luggage cart. Its projections on the two sets of rotated basis shapes is shown in Figure (1.9)(b). The plots in Figure (1.9) can be used to distinguish between the two activities, given just their motion trajectories by setting an appropriate threshold and declaring an activity to be either one or two, depending on the number of points on either side of the threshold. We can thus automatically verify whether each of the different tasks, like passengers boarding a plane or luggage loaded into the cargo hold and the cart departing, were completed successfully or not.

The next task is to determine any abnormalities. By this we mean the detection of the case shown in Figure (1.7)(a). Since the testing is done for each object at a time, the process can identify the concerned individual or object. As we do not have real video sequences of such



(a) (b)

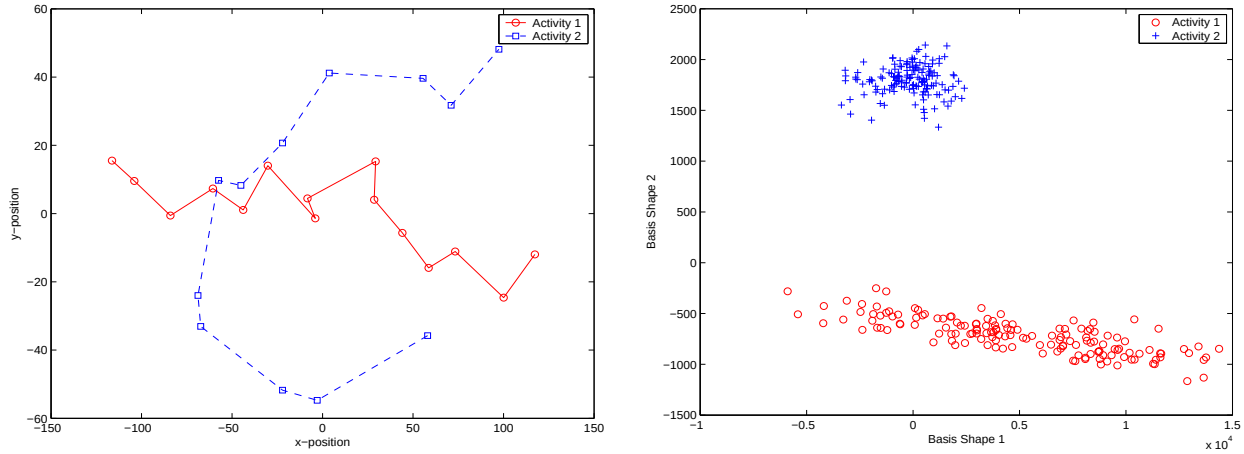
Figure 1.7: (a): An example of an abnormal activity where the average trajectory is distorted to simulate an abnormal behavior. (b): Projections of the abnormal activity and a normal one on the rotated basis shapes for the first activity.

behavior, we simulated it by pulling a passenger away from the normal path. Figures (1.7)(b) plots the projections for the abnormal activity and a normal one on the set of rotated basis shapes. The clear difference in the projections shows the difference in the two activities, which can help to identify the abnormal one.

The Receiver Operating Characteristic (ROC) of the activity detection algorithm, is shown in Figure 1.10(a). The plots are obtained through simulations by varying the threshold of detection for the two normal activities, as well as the abnormal one. For classification between the two activities, a detection occurs when a test activity, say A, is recognized correctly from the projections onto the set of rotated basis shapes of A, while a false alarm is defined as the case when the projections onto the rotated basis shapes of A of the trajectory obtained from some other activity exceeds the detection threshold. For an abnormal activity, a detection occurs when it is correctly identified as abnormal, while a false alarm occurs when a normal activity is flagged as abnormal.

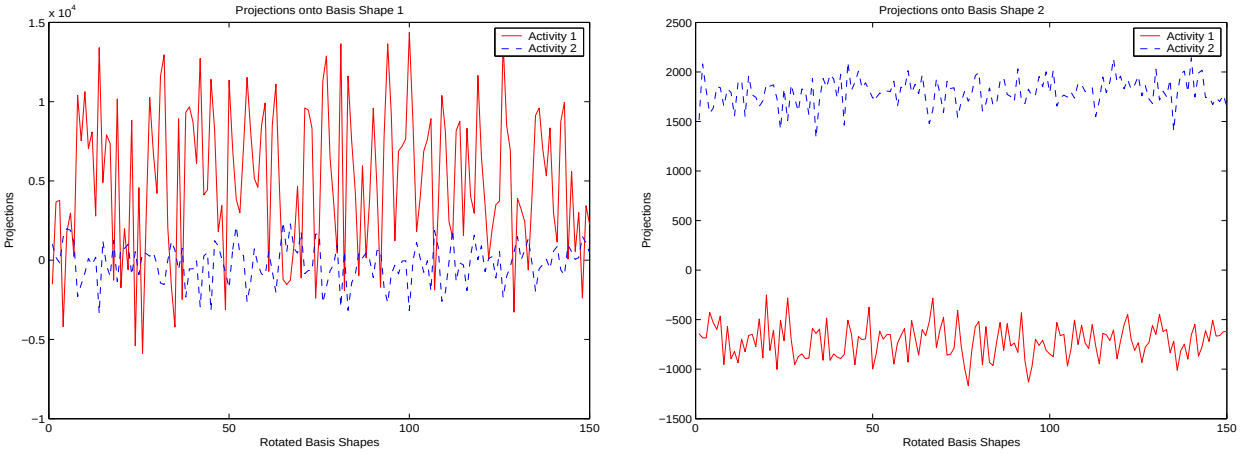
1.5.4 Video Summarization in Shape Space

We performed an experiment to summarize a three minute segment of video obtained for the airport surveillance example in the activity shape space using the subspace analysis method. The motion trajectories of all moving objects were considered. They included the passengers, a luggage cart



(a) (b)

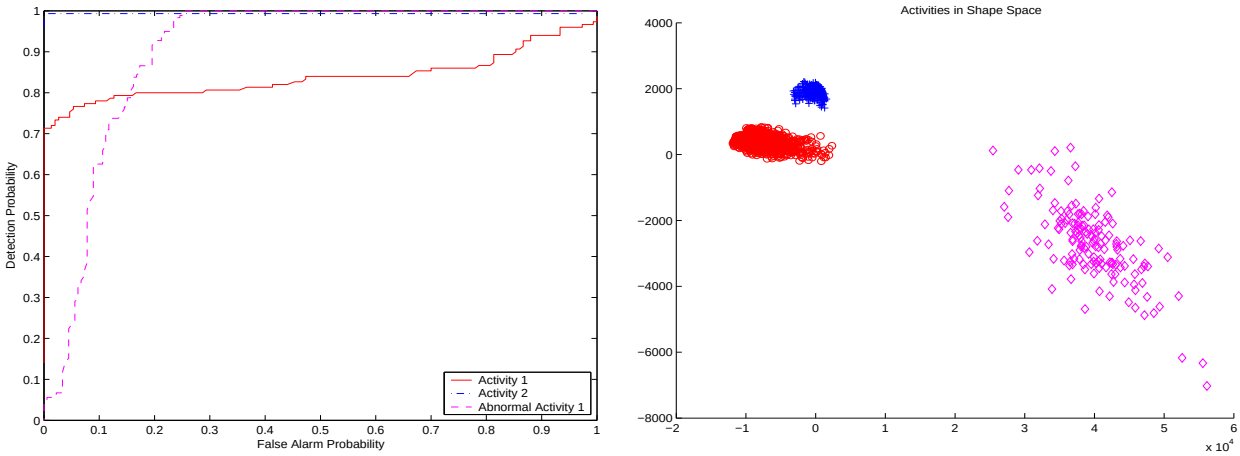
Figure 1.8: (a) Plot of the centered shapes formed from the average trajectories of the two activities. (b) Plot of the projections of the various instances of the two activities, as available in the training data, onto the rotated basis shapes.



(a) (b)

Figure 1.9: Projections of the two activities on the rotated basis shapes for the first one are shown in (a), while the projections on the rotated basis shapes for the second one are shown in (b).

and an airport personnel (whose motion has not been modeled as part of the training procedure, but who can be seen at the bottom of Figure 1.2(b)). The motion trajectory of each individual object was projected onto the set of rotated basis shapes $\mathbf{R}_m S_i$, for $m = 1, ..., 150$, $i = 1, 2$, learned from the training examples, as explained before. Figure 1.10(b) shows the projections form three clusters, corresponding to the motion trajectories of 10 passengers, the luggage cart and an airport personnel. These three clusters contain information about all the moving objects in the three-minute segment of the video. The clusters can also be useful for identifying an abnormal activity,



(a) (b)

Figure 1.10: (a): ROC plots for classification of the two normal activities and the abnormal one. (b): A video summarization example: projections of all the motion trajectories in a three minute segment of the video sequence onto the basis shapes. The red cluster contains the projections of the passengers, the blue of the luggage cart and magenta of the airport personnel whose motion was not modeled as part of the training examples.

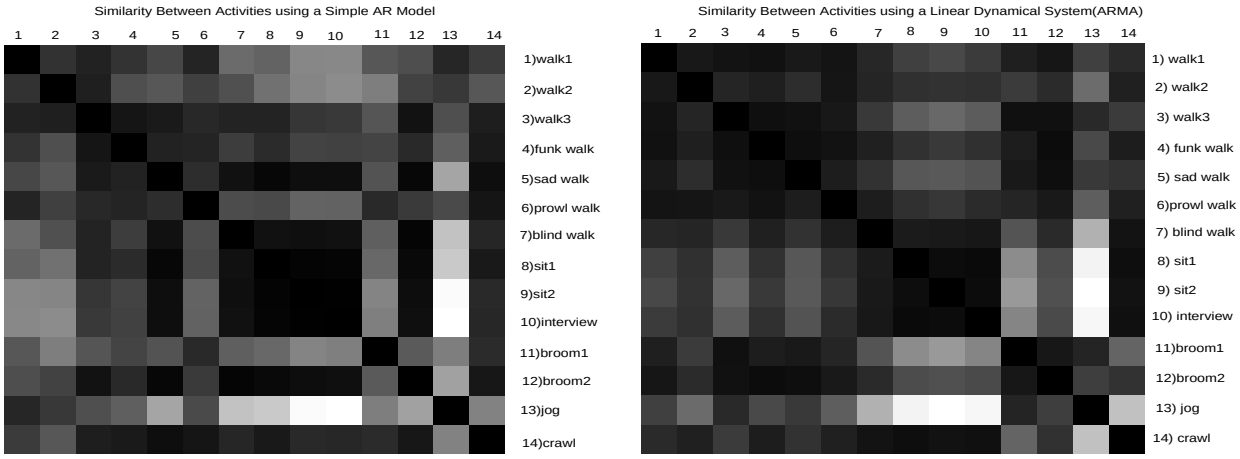
which does not lie in any of the clusters learned for the set of normal activities. Hence we see that it is possible to summarize the motion of all objects in the scene in the shape space.

1.6 Discussion: Are 3D Models Required?

A valid question that can be raised is the following: Do we need to build 3D models of shape, which are often not easy to obtain, is order to perform activity classification accurately? We have performed extensive experiments to understand the role of shape and dynamics in human activity inference. A separate paper on this issue is available [85]. We will quote two results from that paper in order to experimentally justify the use of 3D models.

Consider the vector of points representing the activity in each frame to be $\alpha(t)$, $t = 1, ..., F$. First we consider an autoregressive (AR) model on these points, i.e. $\alpha(t) = A\alpha(t-1) + w(t)$, w is a zero mean white Gaussian noise process and A is the transition matrix. If A_j and B_j (for $j = 1, 2, ...N$) represent the transition matrices for two sequences representing two activities, then the distance between models is defined as $D(A, B)$

$$D(A, B) = \sum_{j=1}^{j=N} ||A_j - B_j||_F \tag{1.16}$$



(a) (b)

Figure 1.11: Plot of the similarity matrix for activity classification using (a) AR models, (b) ARMA model.

where $||.||_F$ denotes the Frobenius norm. The model in the gallery that is closest to the model of the given probe is chosen as the identity of the person.

Next an autoregressive moving average (ARMA) model on the points is used. This linear dynamical model can be represented as

$$\alpha(t) = Cx(t) + w(t); w(t) \sim N(0, R) \tag{1.17}$$

$$x(t+1) = Ax(t) + v(t); v(t) \sim N(0, Q). \tag{1.18}$$

Let the cross correlation between the observation and system noise, w and v, be given by S. The parameters of the model are given by the transition matrix A and the state matrix C. We note that the choice of matrices A, C, R, Q, S is not unique. But, we also know [54] that we can transform this model to the "innovation representation". The model parameters in the innovation representation are unique. The model parameters are learned using the algorithm is described in [54] and [72]. The distance between two ARMA models, ($[A_1, C_1]$ and $[A_2, C_2]$, is computed using subspace angles [27] and as described in [17]. More information on this process can be found in [85].

The plots of the similarity matrices for the activities in the MOCAP data using the AR

and ARMA models are shown in Figure 1.11. Note that the AR model uses purely dynamical information, while the ARMA model encodes 2D shape information also in the C matrix. Comparing these two figures leads to the following conclusion: A pure dynamical model (AR) has less discriminating power than an ARMA model. For example, all the walk sequences in Figure 1.11(a) are not grouped together, as they should be. However, comparison of these two figures with Figure 1.5(b) shows that the use of the 3D model increases the recognition performance even when there is no change in viewing direction. For example, the similarities between jogging and walking are clearer than when using an ARMA model. Also, crawling is clearly distinct from the rest in Figure 1.5(b). Hence there is a clear advantage in using 3D models over 2D models for activity classification. In addition, 3D models allow view invariant recognition and multi-camera use, as we have explained before.

Chapter 2

ONTOLOGY DRIVEN ACTIVITY MODELLING AND RECOGNITION

2.1 Introduction

2.1.1 What is Ontology?

Thomas Gruber defines ontology in artificial intelligence as an explicit specification of a conceptualization [29]. The term "*ontology*" has its roots in philosophy, in which it is defined as the study of being, a system in which the nature and relations of being are explored. We can say that it is a systematic representation of our knowledge about the particular system we are expressing.

It is important to distinguish ontology and taxonomy, which are commonly confused with each other. Taxonomy, having biological origin, is categorization of concepts, in which each component in the tree is placed on a hierarchical tree. In ontology many distinct relations can be defined in order to represent the relations between concepts in the real world more effectively. For instance a car can have a "*driven_by*" relationship with human beings. At the same time the same car can have a "*built_by*" relationship with a company, which is run by human beings.

2.1.2 What can ontology provide?

Ontology provides detailed reasoning among its components. There are various terms used in systematization of concepts. The most common term is lexicon, which is simply a list of concepts, without any explicit specification of relationships between its elements. In taxonomy these relationships are hierarchical relationships, hence limited reasoning can be applied which mainly verifies whether an ancestral relation exists between given 2 elements. For instance we can say that mouse is an animal as it is a mammal and mammals are animals. On the other hand ontology can have as many relations as necessary to effectively model the concepts necessary, hence the relations are not limited to hierarchical ones. Also ontology can be used to put constraints over

its elements. Human beings have the ability to run and walk but do not have the ability to fly without help of an additional device. Hence ontology can be used to build a consistent knowledge base with logical restrictions.

2.1.3 Ontology Evaluation

Gruber defines 5 important criteria for ontology design as clarity, coherence, extendibility, minimal encoding bias and minimal ontological commitment [29]. Jones et al define 4 different metrics for ontology evaluation, namely syntactic quality, semantic quality, pragmatic quality, and social quality. Each metric has individual attributes. Consistency and clarity are included as attributes of semantic quality. Pragmatic quality has attributes for amount, accuracy of information and its relevance for a given task. After scoring for individual definitions they calculate the overall quality by a weighted average (depending on application) of these metrics [6].

2.1.4 Current usage and tools for ontology

Ontology research has been a concern for AI community, which recently became much more popular with the concept of semantic web for increased search efficiency on the web with usage of meta-data. Recently OWL Web Ontology Language (an XML based language on editing ontologies) is recommended by W3C consortium. There are several ontology editors for OWL, particularly Protege is a well-supported one among others [1].

2.2 Systematic metadata in visual environments

Systematic metadata can be used in various steps of automation for reliable detection and efficient search of visual data.

2.2.1 Image Search

One fundamental usage is on image search. For instance in google, image search is based solely on textual search the path and filename for the image file, which causes all irrelevant poses of street fighters and wrestlers when we are looking for fighter planes in 2nd world war and type in keyword

"fighter". It depends on the users ability and luck to guess the naming scheme to have quality results.

In library of congress, images are annotated by hand, and they use a system called TGM (Thesaurus for graphic material) for retrieval on their searches [2]. Although they called it thesaurus, their structure is taxonomy of words with a hierarchy having links to a broader ancestor and narrower subcategories, and an additional list of links to other related words in a separate part. However the relations of the word to the ones on the related list are not classified. Rada et al [64] assigned weights to different relationships in the thesaurus trying to imitate human assessment in order to improve the search quality. Kim and Kim [42] improved on their work to use the hierarchical structure of the thesaurus more effectively. Haase and Tam worked on ways to search through concepts rather than keywords themselves [31]. The system they built outputs users concepts that are related to their keywords, and then the user can manually narrow down to the concept he is interested. Yet for this to be of practical use, there should be a fine granularity large hand built database of concepts with corresponding images which is too costly to be done for a field like search with infinite possibilities.

2.2.2 Video Search

Another important usage for metadata is in video search, mainly used by news broadcasters like CNN for broadcasting and documentaries; and intelligence agencies. For CNN, annotation is done by hand and they have a huge dictionary with 400.000 words that expands with each annotation that contains a word not added before. Yet this kind of a search is only capable of retrieving what individual annotator sees and agreement of the terms of user with that of annotator while doing the search.

2.2.3 Video Markup

As search becomes increasingly important, also annotation of multimedia becomes vital. There is a high need for robust reliable automated annotation systems, and having a metadata structure is important in order to have a unique basis on research for automated video markup. Currently,

individual reporters do video annotation by hand for the broadcasting firms. Automation of the video markup could provide much use, yet it is very hard without any limitation on the context. This is why there has been extensive research on annotation tools that would assist user better, models that would enhance consistency between annotators and the end users, and systems that would enhance annotation with combining information/annotations from various annotators [4, 20, 19, 52, 89].However within a predefined context there has been work on automated annotations. Bertini et al[9] provided an examplary usage of automated annotation in soccer videos, and used this to more effectively compress the videos with higher loss rates in the parts that would be of less importance for the audience. The decision for the importance is decided by the results of automated annotation, which uses an ontology structure and ranks zones of important events like playfield or zone around the goal box. Also Vetro et al worked on object-based transcoding framework that uses relevant information for archival of long term video surveillance data[3].

2.2.4 Video Surveillance

An important aspect of security is usage of video surveillance. Security cameras are located in places that are of strategic interest for crime, vandalism and terrorist activities. However it is costly to assign human labor for each camera placed in these locations. Usage of automated suspicious event detection systems could greatly reduce the cost for security of these areas. Although there are currently no known commercially available automated detection systems to the best of our knowledge, continuous research is going on in this field. Here metadata is essential for formalizing the way to detect suspicious activities and uniting the research going on for automation of the activity recognition in the given contexts. However usage of taxonomy will not be effective in activity recognition and event detection. Because detection of sub-events is necessary to detect a composite event, and there are complex relations within those sub-events with different constraints. For instance if someone is tailgating, his entrance should follow an entry of another person authorized to enter the area (temporal constraint), and he should be hiding from the person entering before him (spatial constraint). Because of these complex relations, if there is need for a

systematic metadata to detect suspicious events, an ontology structure is more suitable.

As the research in automation of video surveillance matures, the need for uniting the research for detection of components needed for larger events is accentuated. Without an agreement, published papers will be results taken with different videos and different contexts which lack coherence. Hence research towards complete automated surveillance systems could be much improved with introduction of a standardization. "Ontologies provide a way to establish common vocabularies and capture domain knowledge for organizing the domain with a community wide agreement or with the context of agreement between leading domain experts."[46].

2.3 Ontology in video surveillance systems

Ontology has been recently used in different contexts for video surveillance. Chen et al used ontology for analysing social interaction in nursing homes[16], Hakeem and Shah used ontology for classification of meeting videos [32].

Ontology examples given in the thesis are from the outcome of ARDA Video Event Challenge Workshop in La Jolla, CA, at December 2003. During the workshop ontology is produced for 6 domains of video surveillance: Perimeter and Internal Security, Railroad Crossing Surveillance, Visual Bank Monitoring, Visual Metro Monitoring, Store Security and Airport-Tarmac Security. As output of that workshop two formal languages were developed. First one is VERL(Video event representation language) which is an ontological representation of complex events in terms of simpler subevents. The second one is VEML (Video Event Markup Language), which is used to annotate VERL events in the videos[63].

In this section we want to examine ARDA workshop output, evaluate the ontology for different domains in terms of ontology principles, show their strengths and weaknesses, and give examples about how these weaknesses can be improved. Let's first give the exact definitions for those concepts:

Clarity: An ontology should effectively communicate the intended meaning of defined terms with

complete formalism.

Coherence: An ontology should have inferences that are consistent with the definitions. At the least, the defining axioms should be logically consistent.

Extendibility: An ontology should offer a conceptual foundation for a range of anticipated tasks,so that one can extend and specialize the ontology monotonically.

Minimal encoding bias: The conceptualization should be specified at the knowledge level without depending on a particular symbol-level encoding.

Minimal ontological commitment:An ontology should make as few claims as possible about the world being modeled, allowing the parties committed to the ontology freedom to specialize and instantiate the ontology as needed.

For more detailed definitions of clarity, coherence, minimal ontological commitment, minimal encoding bias, and extendibility reader is again referred to [29].

2.3.1 Clarity in Temporal Relations

In Perimeter and Internal Security Ontology, tailgating is defined as:

SINGLE-THREAD(**tailgate**(ent x, ent y, facility f),
AND(portal-of(entrance, f)),
Sequence(AND(approach(x, y), behind(x,y))
tail-behind(x, y),
get-access(y, entrance),
enter(y, facility),
NOT(get-access(x, entrance)),
enter(x, facility))))

Here we see multiple problems. First problem is that this sequence is not necessary for a tailgating event. "tail-behind" sub-event does not need to be before get-access "sub-event". We will focus on this more in the minimal commitment section. Now let's examine another important problem. There is no concept of time in this domain other than sequentiality. For some events sequentiality is enough to represent the varieties that the event can have, yet there are some events that need more complex temporal relations for coherent and complete conceptualization. Throughout Perimeter and Internal Security ontology, and also "Event Ontology for Store Security" temporal relations are limited to prefix "sequence". Here the "tail-behind" activity should be an event that is occurring before "enter" sub-event for entity x, and enter event for x should be between the time y gets access to the entrance and the portal f for the entrance is closed back.

Of course for all this we need the definitions for before and between in temporal relations. For these definitions and others that are common and important in the concept of time, we can use Ontology of time created by Hobbs et al[**?**]. This well documented ontology helps us get rid of problems like synchronization and unification in temporal relationships.

A good example for application of this principle in terms of temporal coupling is seen in TSA Tarmac security ontology, in fueling event:

Fueling

physical objects:

((**v1:** fuel carrier vehicle)**,(z1:** airplane)**,(eq1:**fuel tank opening of plane)**, (eq2:**fuel pump of carrier))

components:

((**c1:approach(v1,z1))**

(c2:open(eq1))

(c3:inside_of(eq2,eq1))

(c4:together(v1,z1))

(c5:close(eq1))

(c6:leave(v1,z1)))

constraints:

(**sequence(c1,c4,c6)**

(c2,c3,c5 during c4)

sequence(c2,c3,c5))

We see that here, the event c4 that corresponds to the fact of fuel vehicle being near the plane, can not be put in any order with activities c2,c3 and c5 as it is concurrent with those processes.

2.3.2 Clarity of Negation Concept in Time

In the tailgating event 2.3.1 we see two types of prefixes. Most prefixes like get-access, enter, tail-behind,... are prefixes with temporal correctness. Entering event is only true for a given period of time while the entity is entering the zone, but not in other time periods. Also there are time-independent prefixes like portal-of, which represents that the entrance to the facility f is defined by entrance. This is an assertion that is true independent of the time. And for that prefix we do not need to give a time interval for it to be meaningful. This situation is accentuated more causing an ambiguity that needs to be solved when negation is used. For example let's have a look at shoplifting event from Store Security ontology.

SINGLE-THREAD(**shop-lift** (person x, employee y, ent o),

AND(counter(area),

merchandise(o),

Close(area)

NOT present(y,area)

Sequence(move(x,area),

Open(area)

Pick-up(x,y));

Here is an ambiguity about the negation as the employee y not being present in the area is only true for a temporal period. It is not a truth that is independent of time. Hence for this negation there should also be a definition for a time interval. And the time-dependent definition for not should be introduced for time dependent prefixes. We can define it as:

NOT_IN_INTERVAL(*prefix*(*entity_list*),*time_interval*)

meaning that the *prefix*(*entity_list*) event is never true for any sub-interval in the given time interval. By separating time-dependent negation from negations expressing falseness of a concept, we deal with ambiguity caused by negations that are only true for an assumed period. There is an interesting combination that is possible here. If we try to understand meaning of time-independent NOT of a NOT_IN_INTERVAL,i.e.:

NOT(NOT_IN_INTERVAL(*prefix*(*entity_list*),*time_interval*))

, we see that it corresponds to negation of the prefix event not being true in any sub-interval in the given time interval. In other words, it means that the prefix event is true in at least one sub-interval in the time interval, hence the prefix is realized in a subset of the time interval. Once that is clear, the negation concept will not damage the clarity of the ontology anymore. This is a common mistake that is repeated in each ontology in this workshop that uses negation.

2.3.3 Minimal Ontological Commitment

Minimal ontological commitment is another ontological principle that is violated in ontology for most of the domains. A detailed examination for dense violation of this principle in banking scenario is explained in 2.4.2, we will examine other less severe violations of this principle in this section.

One example from perimeter security is:

SINGLE-THREAD(**suspicious-load**(vehicle v, person p, ent obj, facility fac),

AND(zone(loading-area),

near(loading-area, facility),

portable(obj),

Sequence(*approach*(v, fac),

AND(stop(v), *near*(v, fac), NOT*(inside*(v, loading-area))),

AND(approach(p, v), *carry*(p, obj)),

AND(stop(p), *near*(p, v)),

cause(p, open(portal-of(v))),

enter(obj, v),

cause(p, close(portal-of(v))),

leave(v, facility))))

For instance minimal ontological commitment is not preserved in suspicious load. First of all, according to this definition, for it to be a suspicious load, portal of the vehicle has to be opened in order to load the object. Yet this will cause missing of some of suspicious loads as opening of the portal is not a basic component that is necessary for this event. The suspicious load can be placed onto the trailer of a truck which is open from the top, hence not using any portal, or it can even be a bomb that is placed under the body of the vehicle. Moreover, it is not necessary that the vehicle stops if we want to take it a little bit towards the extremes. somebody inside the vehicle can fastly grab a bag from other suspicious person through the window or even from hand to hand while in a motorcycle. Hence minimized ontology should only include the object being outside of the vehicle, and then being transferred to inside of the vehicle while it is in an undesignated zone. And this is minimal to characterize all suspicious load events, as these are the basic components that form this event.

Another less serious violation example can be given from both perimeter security and TSA tarmac

security, in the definiton of process approach:

PROCESS(**approach**(ent x, ent y),

cause(x, change(far(x, y), near(x, y))))

According to definition of approach, the event ends with entity x being near entity y. However in the TSA and perimeter ontology, usages of approach(x,y) is followed by near(x,y), which will not introduce any problems in terms of detection with vision systems but rather will form an unnecessary repetition in ontology.

2.3.4 Unified Representation

Another main problem is with the representation of the ontology for these 6 domains. It is directly seen that they have very different formats from each other. It is important to have the same format for representation of events in order to be able to examine and compare ontology for different domains better. We suggest usage of format from TSA scenario and the banking scenario as they both have labels for individual subcomponents of events, and hence it is easier (and more importantly uniquely possible) to represent the temporal relations between different subcomponents of an individual event.

2.4 Application example of ontology for two different complexity domains

2.4.1 Overview

In this chapter of the thesis we will focus on the role of the context in determination of the necessity for the ontology and required ontology complexity. 2 different domains, namely bank monitoring and TSA (Airport and Tarmac security) are examined with a comparison done for reasoning of ontology usage.

2.4.2 Bank Scenario

In the bank scenario, if we examine the ontology output of the ARDA workshop, we see that there are many safe attack scenarios having only slight differences between each other. In the single threaded ones there are safe attacks with a single person, in which the path of the attacker changes slightly; or a gate is open or not and these are all resulting in different activities. And also there are multi-threaded activities that include two robbers and various combinations of slight changes for each robber and the gate. The problem with such an approach is that, if we define all these to be separate activities, then we should also consider activities with 3 or more robbers as separate, and finally we will end up having infinite number of possibilities even only for a safe attack. And though this is not complete for the safe attack itself. Because there may be many different cases, to give example, some of the robbers may wait outside the safe and check for people around when the others enter the safe, or even weird scenarios. They may try to enter the safe area they are not successful; they kill the employee and runaway. Moreover there are many different suspicious activities that has to be detected inside a bank. They may not even be robbers yet they may be there for an assassination or terrorist activity, killing everybody not taking a single dollar bill and running away. They may even argue with each other after or before taking money and kill each other reservoir dogs style. To sum up there is a problem with going overly deep and granularity issues. This also conflicts with the minimal ontological commitment criterion from the ontology evaluation part. If we examine all these suspicious scenarios a little bit more carefully, in all those scenarios there is at least one of 2 common behaviors.

i. Either someone is killed

ii. Or there is an unauthorized access to safe (Someone other than the employee)

And these are enough to formulate the suspicious activities in a bank (We may also include stealing from the counter itself for a rather small scale robbery, in this case an additional behavior of taking out a gun will be enough for handling these extra situations). Hence in this situation usage of ontology for different types of safe attack scenarios seems like overkill. The same detection capacity of suspicious events can simply be achieved by using at least one of the behaviors listed that are

possible subcomponents of the overall safe attack scenario.

We implemented a simple usage of an examplary ontology for detection of suspicious events in a bank scenario. As we explained the reasons that Arda Bank Ontology conflicts with minimal ontological commitment, we decided to build a simple small ontology that can be used to determine whether there is a safe attack scenario or not. The application we built using that ontological reasoning is a simple example that is able to distinguish between safe attack scenario and an ordinary "no attack scenario", in which customers deal with their transactions, wander around and leave the bank. The simple ontology for safe attack scenario is:

safe_attack: usage: safe_attack(mo1,z1) physical objects:

((**mo1:**mobile object, **),(z1:**zone**)**)

components:

((**c1:approach(mo1,z1))**

(c2:inside_zone(mo1,z1)))

(c3:leave(mo1,z1)))

(c4:NOT(employee((mo1)))))

temporal constraints:

(**sequence(c1,c2,c3)**)

2.4.3 TSA Scenario

Another domain for the usage of ontology in video surveillance is Tarmac and Airport Security. Again some samples for the video frame and the background-extracted motion-tracking image are given. In this scenario though, when we look at the ontology output for the ARDA video challenge, we see that the ontology is minimal in the sense that only necessary and sufficient activities for each event is written down. For instance for passenger getting on, all needed is for a passenger to approach the airplane, and then go inside. Additional granularity details like whether passenger has some hand luggage and leaves it to the airport employees or takes it with him are ignored, as they are irrelevant with the getting on event of the passenger. And here we cannot also define

suspicious activity in a minimized manner like we did in the banking scenario. In banking scenario unauthorized safe access and/or killing of somebody was enough to decide that it is a suspicious activity. But for an aircraft stakes are much higher, security is a much more important issue for the safety of passengers and possible victims of a terrorist attack if the aircraft is hijacked. It is why everything should be exactly done according to the procedures in airports. If there is at least something not going in exact coherence with the procedure at least a closer look is vital. For instance a passenger might approach the aircraft zone, and then leave in another direction without entering the aircraft. Or he might enter the aircraft and then suddenly leave out of the aircraft and go at some direction. Although this can be because of a simple reason like forgetting something in the terminal, it is an unexpected event. And even if it seems paranoid it has at least a very slight probability that the passenger was a terrorist placing an explosive in the plane or playing with the controls of the plane and leaving the scenery, hence it should be dealt with increased care than regular flight scenarios. Hence in a TARMAC security scenario everything should be in complete consistence with the nature of events occurring normally. Any inconsistency should at least form a warning for the security authorities of the airport. Usage of ontology for detection of suspicious events in this case makes more sense. And the necessity for events occurring in a specific order in spatial and temporal space means that we need constraints for the sub-events for forming composite ones. Thus the usage of ontology for detection of such events in airport security is not only sufficient, but also necessary.

We also implemented a simple scenario for passengers getting in and out of plane with airport tarmac video surveillance data, in order to give an examplary usage of ontology for procedural event detection. We used the ontology from ARDA workshop output for detecting those simple events.

2.4.4 Experimental Results

In the following pages we give out the experimental results for the application of our ontology models in recognition of the activities in real life. On each page we added 6 main frames summa-

rizing the video in terms of the ontological principles each with explanations. There are 4 videos in bank attack scenario in which a bank is being robbed in different ways. After that there are 2 more videos in bank scenario which do not contain any suspicious activity, they are mainly just customers looking at the brochures and getting their work done at the bank. All of those activities were correctly identified as "safe attack", and "no safe attack" situations by the detections in which ontology model was used. We want to thank Monique Thonnat for providing us different bank scenario videos. After that there are 2 more TSA videos that contain passengers getting on to a plane and getting off the plane. Also in both of these videos the activities were correctly detected as passengers getting on the plane and passengers getting off the plane.

2.5 Decision for ontology necessity in a given domain

After all this discussion with examples in a top-down strategy for determining where ontology would be useful in different cases, we should also go bottom-up and define properties of cases where usage of ontology would be essential.

We will put some assertions here followed by further clarification of the meaning and implications of these assertions.

a. The power of ontology over taxonomy is effective representation of various kinds of relationships within a group of subcomponents forming the main component.

Then this means that unless we need further relationships rather than categorization usage of ontology is not necessary. The simplest example for this can be given as military detection of weaponry in large amount of videos. If all we need to detect is simply a tank or anti-aircraft machine gun, then all we need is a lexicon. If we want to generalize all them as a subgroup of weapons and detect them altogether, then taxonomy would be enough.

b. For detection of events usage of ontology is not only sufficient but also necessary.

As events are composed of subcomponents with temporal and spatial constraints relative to each other, usage of taxonomy would not be enough for efficient, complete and robust representation of events.

c. For event detection, the necessary granularity for the ontology is dependent on the context (domain).

Given the example of banking scenario, we were able to specify attack scenarios with a smaller subset of its subcomponents, hence a more specific ontology for these 2 small subcomponents was enough to decide that there was an attack or not.

However for the TSA data, although still unauthorized access to restricted zones has to be and is detected with the ontology, higher level activity recognition for suspicious activities like loitering, unplanned abandoning of the plane by one or more of the passengers during getting on process, etc. have to be also clearly identified with the appropriate constraints. Hence usage of robust ontology here is effective for detection of more complex events.

(a) Robber enters and takes out a gun

(b) Robber comes to counter zone to threaten employee

(c) Employee follows order of the robber to open the safe

(d) They both enter the safe zone

(e) They spend some time enough to take the valuables out

(f) Robber leaves the building

Figure 2.1: Bank attack scenario 1: Robber directly goes to counter zone, takes the employee with him and enters safe zone. After collecting valuables inside the safe he leaves the building.

(a) Customer enters building

(b) Robber comes and pushes the customer away

(c) Employee comes out of counter zone to provide access to safe

(d) Employee opens the safe

(e) Customer runs away while robber enters the safe zone

(f) Robber leaves the building

Figure 2.2: Bank attack scenario 2: Robber directly goes to counter zone, takes the employee with him and enters safe zone. After collecting valuables inside the safe he leaves the building. This is exactly the same with attack scenario 1. Only difference is that there is a customer who runs away as soon as they enter the safe.

(a) Robbers enter the building

(b) Employee moves out

(c) Employee and one of the robbers enter the safe

(d) They get out of the safe, in the mean time, other robber is watching

(e) Robbers get ready for the escape

(f) Robbers leave the building

Figure 2.3: Bank attack scenario 3: 2 robbers enter the building. One of them takes the employee out of counter zone and directly goes to the vault. The other one stays inside the building to watch. After collecting valuables inside the safe they both leaves the building. Although now there are two robbers, still the robbery event itself is realized by the robber who entered safe. If he was not there it would not be counted as a robbery.

(a) Customer waits in counter zone

(b) Robbers enter the building

(c) One robber watchs. The other goes to management zone

(d) Robber uses manager to access management zone

(e) They are out of safe, getting ready to escape

(f) Robbers leave the building

Figure 2.4: Bank attack scenario 4: Two robbers and a customer. One robber waits inside the building for watching and keeping the customer and the counter clerk inside the building. The other one goes to management office, takes out the manager, and uses his access to enter the safe. Still detection of unauthorized access to safe is enough to judge that this is a robbery

(a) (b)

(c) (d)

(e) (f)

Figure 2.5: Bank no-attack scenario 1: One customer looks around to the brochures etc, while the other one is having his job done by the clerk

(a) (b)

(c) (d)

(e) (f)

Figure 2.6: Bank no-attack scenario 2: Another no attack scenario. Very similar to the previous one. In both of these scenarios, there is no unauthorized access to safe

(a) First passenger appears in entry zone

(b) Passenger leaves the entry zone and approaches plane zone

(c) Other passengers start flowing through the route while first one enters plane zone

(d) Passenger get on activity detected, other passengers are on their ways to get on the plane

(e) Last passenger moves to get on the plane

(f) Last passenger gets on the plane

Figure 2.7: TSA scenario passengers getting on the plane: The expected procedure is exactly followed, passengers come through entrance area, they approach the plane zone and they get on.

(a) First passenger appears out of plane zone

(b) Others start following the first passenger

(c) Get off activity continues while some pack luggages they got from the chart

(d) Get off detection is complete, as already a few passengers got off. The other passengers are in different stages of get off activity

(e) Last passenger is also out of the plane on his way towards exit

(f) All of the remaining passengers are approaching exit zone

Figure 2.8: TSA scenario passengers get off: Regular procedure of passengers getting off the plane. Some of them take their luggages outside, yet it is not a basic component of getting off activity as people can simply get off even when they don't have luggage. Yet clearly the overall procedure can simply be represented by ontological relations shown here.

d. Ontology should have minimal commitment and be optimized for the specific task given.

These are both highlighted also in ontology evaluation before in this thesis. However, here optimization is much more important than other aspects. Because the main problem in automated detection systems is detection and tracking of components. There already are enough problems in these systems like noise, intensity changes, occlusion, etc. If we try to use a generic ontology for tracking everything in various video surveillance systems like railroads and airports, we would have more choices of detection for a vehicle to be a small train or a luggage truck. Of course each one makes much more sense in their respective domains. As the video surveillance system is a stable system regarding the fact that it is theoretically built to stay in its original position forever, there is no fault in optimizing the system for its own view. Still there should be some common activities in both domains like people walking, running, etc. Yet these are already grouped under video surveillance common activities section for these 6 domains in ARDA video challenge workshop anyway. The importance of ontology is on agreement, definition and clarification of the concepts. Once these concepts are agreed on, it would be much easier to produce specific automated systems for individual domains and increase their effectiveness with ongoing research over a common ground.

2.5.1 Event detection after ontology

After the ontology is designed, then there comes the necessity of detecting events using the model we have. Individual sub-parts of the ontology with spatial constraints (like being close to safe zone, or being inside the tarmac zone) can be detected in the video simply by tracking and background extraction. Once those individual sub-parts are determined, the event with temporal constraints is formed by a sequential order of those sub-parts within a time interval. Regular expressions are enough to describe such relationships and DFAs can be used to detect those events. As multithreading is possible (combination of different events simultaneously), a state for each DFA should be kept every moment. If one of the detected sub events is the beginning transition for

one of the activities, then corresponding state can be changed. Here, also the importance of ontology minimalism is accentuated, as the DFAs for a non-minimal ontology would result in missed detections and ambiguities for separate events, damaging the robustness of the overall system.

Chapter 3

Conclusion and Future Work

In this study, we proposed two different methodologies for activity modeling and recognition.

In the first part we have proposed a method for activity modeling and inference using 3D deformable shape models representing the configuration of points taking part in the activity. The 3D shape is estimated from the motion trajectories of the points under the assumption of weak perspective projection. The approach fits into the general framework of inferring high level information about different activities starting from the trajectories. Our approach is independent of the viewing direction of the camera and can be extended to the situation of a video sensor network looking at the scene. We have also proposed a method for estimating the amount of deformation in a shape sequence, terming it as the "deformability index". This is used in the estimation of the 3D shape models. Experimental results are shown for classifying between various human activities like walking, jogging, sitting, etc., as well as for the activities of a group of people in an airport surveillance scenario.

In the second part, ontology is examined as a structure of metadata. Usage of metadata for different fields of vision are discussed, and then it is decided that the usage of ontology is not needed for image search without subjective reasoning necessity, and it may be an overkill for applications that do not need an event detection scheme. For instance in video search if the objective is to detect a particular type of object (tank, weaponry, particular background, etc.) throughout videos, than taxonomy should be enough to model the metadata essential for the search. Then the usage of ontology in video surveillance is focused on. It is explained that it is necessary to use ontology whenever there is a need to effectively detect events. It is shown that whether the granularity of the ontology in a video surveillance system is dependent on the context of surveillance data. 2 different contexts are examined to give a better example, bank monitoring and tarmac security. It is shown that the necessity of higher-level ontology for detection of suspicious activities is highly

dependent on the procedural nature of the events occurring in the given domain. Hence although it is not necessary to use higher-level granularity ontology for bank monitoring, for tarmac security it is an important and necessary idea for correctly modeling suspicious event detection. As soon as the ontology for individual domains are finalized and some standardization is provided then individual research areas for automated surveillance detection will have a common ground to work on, so that isolation and improvement can be done in a synchronized and growing fashion. Once the requirements are absolutely clarified, it would also be much more convenient to effectively evaluate and compare the efficiency of various systems for a standardized purpose. It is important to have a unified direction for the research with a common ground, otherwise all we will end up is a bunch of random walks that add up to only slightly further than where we stand for a large amount of effort.

BIBLIOGRAPHY

[1] Web link for protege download and support, Stanford University, http://protege.stanford.edu/.

[2] Thesaurus for graphic material search page, Library of Congress, http://lcweb.loc.gov/rr/print/tgm1.

[3] K. Sumi A. Vetro, T. Haga and S. H. Object-based coding for long-term archive of surveillance video. In *IEEE Conference on Multimedia and Expo*, pages 417–420, 2003.

[4] Gregory D. Abowd, Matthias Gauger, and Andreas Lachenmann. The family video archive: an annotation and browsing environment for home movies. In *MIR '03: Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*, pages 1–8. ACM Press, 2003.

[5] K. Akita. Image sequence analysis of real world human motion. *Pattern Recognition*, 17:73–83, 1984.

[6] V. Sugumaran Andrew B. Jones, Veda C. Storey and P. Ahluwalia:. Assessing the effectiveness of the daml ontologies for the semantic web. June 2003.

[7] D. Ayers and R. Chellappa. Scenario recognition from video using a hierarchy of dynamic belief networks. In *Proc. of Intl. Conf. on Pattern Recognition*, pages 835–838, 2000.

[8] A.M. Baumberg and D.C. Hogg. An efficient method for contour tracking using active shape models. In *TR*, 1994.

[9] Marco Bertini, Alberto Del Bimbo, Rita Cucchiara, and Andrea Prati. Semantic video adaptation based on automatic annotation of sport videos. In *MIR '04: Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pages 291–298. ACM Press, 2004.

[10] C. Bregler. Learning and recognizing human dynamics in video sequences. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 568–574, 1997.

[11] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 8–15, 1998.

[12] B. F. Bremond and M. Thonnat. Analysis of human activities described by image sequences. In *Proc. Intl. Florida AI Research Symp.*, 1997.

[13] H. Buxton and S. Gong. Visual surveillance in a dynamic and uncertain world. *Artificial Intelligence*, pages 431–459, 1995.

[14] Q. Cai and J.K. Aggarwal. Tracking human motion using multiple cameras. In *Proc. of Intl. Conf. on Pattern Recognition*, pages C: 68–72, 1996.

[15] C. Castel, L. Chaudron, and C. Tessier. What is going on? a high-level interpretation of a sequence of images. In *ECCV Workshop on Conceptual Descriptions from Images*, 1996.

[16] Datong Chen, Jie Yang, and Howard D. Wactlar. Towards automatic analysis of social interaction patterns in a nursing home environment from video. In *MIR '04: Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pages 283–290. ACM Press, 2004.

[17] K.D. Cock and D.B. Moor. Subspace angles and distances between arma models. *Proc. of the Intl. Symp. of Math. Theory of networks and systems*, 2000.

[18] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models: Their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, January 1995.

[19] Nuno Correia and Teresa Chambel. Active video watching using annotation. In *MULTIMEDIA '99: Proceedings of the seventh ACM international conference on Multimedia (Part 2)*, pages 151–154. ACM Press, 1999.

[20] Miguel Costa, Nuno Correia, and Nuno Guimarães. Annotations as multiple perspectives of video content. In *MULTIMEDIA '02: Proceedings of the tenth ACM international conference on Multimedia*, pages 283–286. ACM Press, 2002.

[21] T.J. Darrell and A.P. Pentland. Space-time gestures. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 335–340, 1993.

[22] J. Davis and A. Bobick. The representation and recognition of action using temporal templates. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 928–934, 1997.

[23] C. Dousson, P. Gabarit, and M. Ghallab. Situation recognition: Representation and algorithms. In *Proc. Intl. Jt. Conf. on AI*, pages 166–172, 1993.

[24] W. Freeman. Computer vision for television and games. In *RATFG99*, pages xx–yy, 1999.

[25] D.M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, January 1999.

[26] D.M. Gavrila and L.S. Davis. 3d model-based tracking of humans in action: A multi-view approach. In *UMD*, 1995.

[27] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1989.

[28] W.E.L. Grimson, L. Lee, R. Romano, and C. Stauffer. Using adaptive tracking to classify and monitor activities in a site. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 22–31, 1998.

[29] Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. *Int. J. Hum.-Comput. Stud.*, 43(5-6):907–928, 1995.

[30] Y. Guo, G. Xu, and S. Tsuji. Understanding human motion patterns. In *Proc. of Intl. Conf. on Pattern Recognition*, pages B:325–329, 1994.

[31] Ken Haase and David Tamés. Babelvision: better image searching through shared annotations. *interactions*, 11(2):18–26, 2004.

[32] Asaad Hakeem and Mubarak Shah. Ontology and taxonomy collaborated framework for meeting classification. In *ICPR (4)*, pages 219–222, 2004.

[33] G.F. Harris and P.A. Smith (Editors). *Human Motion Analysis: Current Applications and Future Directions*. IEEE Press, 1996.

[34] S. Hongeng and R. Nevatia. Multi-agent event recognition. In *Proc. of International Conf. on Computer Vision*, pages II: 84–91, 2001.

[35] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber. Automatic symbolic traffic scene analysis using belief networks. In *Proc. AAAI*, pages 966–972, 1994.

[36] S. Intille and A. Bobick. A framework for recognizing multi-agent action from visual evidence. In *Proc. AAAI*, pages 518–525, 1999.

[37] S. Ioffe and D.A. Forsyth. Human tracking with mixtures of trees. In *Proc. of International Conf. on Computer Vision*, pages I: 690–695, 2001.

[38] G. Johansson. Visual perception of biological motion and a model for its analysis. *PandP*, 14(2 1973):201–211, 1973.

[39] R.E. Kahn, M.J. Swain, P.N. Prokopowicz, and R.J. Firby. Gesture recognition using perseus architecture. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 734–741, 1996.

[40] I.A. Kakadiaris and D. Metaxas. Model-based estimation of 3d human motion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(12):1453–1459, December 2000.

[41] A. Kale, A.N. Rajagopalan, A. Sundaresan, N. Cuntoor, A. Roy-Chowdhury, A. Krueger, and R. Chellappa. Identification of humans using gait. *IEEE Trans. on Image Processing*, pages 1163–1173, September 2004.

[42] Young Whan Kim and Jin H. Kim. A model of knowledge based information retrieval with hierarchical concept. *J. Doc.*, 46(2):113–136, 1990.

[43] Y. Kuniyoshi and H. Inoue. Qualitative recognition of ongoing human action sequences. In *Proc. Intl. Jt. Conf. on AI*, pages 1600–1609, 1993.

[44] S. Kurakake and R. Nevatia. Description and tracking of moving articulated objects. In *Proc. of Intl. Conf. on Pattern Recognition*, pages I:491–495, 1992.

[45] D. Metaxas and D. Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(6):580–591, June 1993.

[46] John A. Miller, Gregory T. Baramidze, Amit P. Sheth, and Paul A. Fishwick. Investigating ontologies for simulation modeling. In *ANSS '04: Proceedings of the 37th annual symposium on Simulation*, page 55. IEEE Computer Society, 2004.

[47] H. Moon, R. Chellappa, and A. Rosenfeld. 3d object tracking using shape-encoded particle propagation. In *Proc. of International Conf. on Computer Vision*, pages II: 307–314, 2001.

[48] G. Mori and J. Malik. Estimating human body configurations using shape context matching. In *Proc. of European Conference on Computer Vision*, 2002.

[49] R.M. Murray, Z. Li, and S.S. Sastry. *A Mathematical Introduction To Robotic Manipulation.* CRC Press, 1994.

[50] E. Muybridge. *The Human Figure in Motion.* Dover Publications, 1901.

[51] H. Nagel. From image sequences towards conceptual descriptions. *Image and Vision Computing*, pages 59–74, 1988.

[52] Marc Nanard and Jocelyne Nanard. Cumulating and sharing end users knowledge to improve video indexing in a video digital library. In *JCDL '01: Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries*, pages 282–289. ACM Press, 2001.

[53] B. Neumann and H.J. Novak. Event models for recognition and natural language descriptions of events in real-world image sequences. In *Proc. Intl. Jt. Conf. on AI*, pages 724–726, 1983.

[54] P.V. Overschee and B.D. Moor. Subspace algorithms for the stochastic identification problem. *Automatica*, 29:649–660, 1993.

[55] V. Parmeswaran and R. Chellappa. View invariants for human action recognition. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2003.

[56] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, 1988.

[57] A.P. Pentland. Automatic extraction of deformable part models. *IJCV*, 4(2):107–126, March 1990.

[58] A.P. Pentland, A. Azarbayejani, N. Oliver, and M. Brand. Real-time 3-d tracking and classification of human behavior. In *DARPA97*, pages 193–200, 1997.

[59] A.P. Pentland and B. Horowitz. Recovery of nonrigid motion and structure. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(7):730–742, July 1991.

[60] P. J. Phillips, S. Sarkar, I. Robledo, P. Grother, and K. W. Bowyer. The gait identification challenge problem: Data sets and baseline algorithm. *Proc of the International Conference on Pattern Recognition*, 2002.

[61] R. Polana and R.C. Nelson. Low level recognition of human motion. In *Non-Rigid94*, pages XX–YY, 1994.

[62] F. Quek, D. McNeill, R. Bryll, C. Kirbas, H. Arslan, K.E. McCullough, N. Furuyama, and R. Ansari. Gesture, speech, and gaze cues for discourse segmentation. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages II:247–254, 2000.

[63] J. Hobbs R. Nevatia and B. Bolles. An ontology for video event representation. In *IEEE Workshop on Event Detection and Recognition*, 2004.

[64] Roy Rada and Ellen Bicknell. Ranking documents with a thesaurus. *JASIS*, 40(5):304–310, 1989.

[65] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 50(2):203–226, 2002.

[66] J.M. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *Proc. of International Conf. on Computer Vision*, pages 612–617, 1995.

[67] P. Remagnini, T. Tan, and K. Baker. Agent-oriented annotation in model based visual surveillance. In *Proc. of International Conf. on Computer Vision*, pages 857–862, 1998.

[68] N. Rota and M. Thonnat. Activity recognition from video sequence using declarative models. In *ECAI 2000*, 2000.

[69] G. Shaffer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[70] T. Shakunaga. Pose estimation of jointed structures. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 566–572, 1991.

[71] A. Shio and J. Sklansky. Segmentation of people in motion. In *MOTION91*, pages 325–332, 1991.

[72] S. Soatto, G. Doretto, and Y.N. Wu. Dynamic textures. *Proc. of International Conf. on Computer Vision*, 2:439–446, 2001.

[73] S. Soatto and A.J. Yezzi. Deformotion: Deforming motion, shape average and the joint registration and segmentation of images. In *Proc. of European Conference on Computer Vision*, page III: 32 ff., 2002.

[74] T. Starner and A. Pentland. Visual recognition of american sign language using hidden markov models. In *Proc. Intl. Workshop on Face and Gesture Recognition*, 1995.

[75] P. Stoica and R. Moses. *Introduction to Spectral Analysis*. Prentice Hall, 1997.

[76] Z. Sun, V. Ramesh, and A.M. Tekalp. Error characterization of the factorization method. *Computer Vision and Image Understanding*, 82(2):110–137, May 2001.

[77] R. Tanawongsuwan and A.F. Bobick. Gait recognition from time-normalized joint-angle trajectories in the walking plane. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages II:726–731, 2001.

[78] R. Tanawongsuwan and A.F. Bobick. Modelling the effects of walking speed on appearance-based gait recognition. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages II:783–790, 2002.

[79] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, 9:137–154, November 1992.

[80] L. Torresani and C. Bregler. Space-time tracking. In *Proc. of European Conference on Computer Vision*, 2002.

[81] L. Torresani, D.B. Yang, E.J. Alexander, and C. Bregler. Tracking and modeling non-rigid objects with rank constraints. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages I:493–500, 2001.

[82] K. Toyama and A. Blake. Probabilistic tracking in a metric space. In *Proc. of International Conf. on Computer Vision*, pages II: 50–57, 2001.

[83] S. Tsuji, A. Morizono, and S. Kuroda. Understanding a simple cartoon film by a computer vision system. In *Proc. Intl. Jt. Conf. on AI*, pages 609–610, 1977.

[84] N. Vaswani, A. Roy-Chowdhury, and R. Chellappa. Activity recognition using the dynamics of the configuration of interacting objects. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2003.

[85] A. Veeraraghavan, A. Roy-Chowdhury, and R. Chellappa. Role of shape and kinematics in human movement analysis. In *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2004.

[86] V.T. Vu, F. Bremond, and M. Thonnat. Automatic video interpretation: A recognition algorithm for temporal scenarios based on pre-compiled scenario models. In *CVS03*, page 523 ff, 2003.

[87] A. Wilson and A Bobick. Recognition and interpretation of parametric gesture. In *Proc. of International Conf. on Computer Vision*, pages 329–336, 1998.

[88] C.R. Wren, A. Azarbayejani, T.J. Darrell, and A.P. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.

[89] Andrew Yao and Jesse Jin. The development of a video metadata authoring and browsing system in xml. In *CRPITS '00: Selected papers from the Pan-Sydney workshop on Visualisation*, pages 39–46. Australian Computer Society, Inc., 2001.