A Real Time System for Detection and Tracking of People and Recognizing Their Activities

Ismail Haritaoglu
Computer Vision Laboratory
University of Maryland
College Park, MD 20742

Abstract

Recently, there has been a movement in computer vision from the processing of static images to the processing of video sequences. Current research has begun to investigate the recognition of human activities taking place in the scene. Applications such as video database, virtual reality interfaces, smart surveillance systems all have in common to track and interpret human activities.

We propose a low-cost PC based real-time visual surveillance system, called W^4, for tracking people and their body parts, and monitoring their activities in monochromatic and stereo imagery. It operates on grayscale video imagery, or on video imagery from an infrared camera. Unlike many systems for tracking people, our system make no use of color cue. Instead W^4 employs a combination of shape analysis, robust tracking techniques, silhouette based body model to locate and track the people and understand the interaction between people and objects - e.g., people exchanging objects, leaving objects in the scene. A subsequent system, W^4S, integrated real-time stereo computation into W^4. Incorporation of stereo has allowed us to overcome the difficulties that W^4 encountered with sudden illumination changes, shadow and occlusion which makes tracking much harder in intensity images. A new silhouette-based body model *Ghost* is described to determine the location of body parts while the people are in generic postures. It is a combination of a hierarchical body pose estimation, a convex hull analysis of the silhouette, and a partial mapping from the body parts to the silhouette segments using a distance transform method that incorporates the topology of the human body. Future research is outlined at the end of proposal.

1 Introduction

Recently, there has been a movement in computer vision from the processing of static images to the processing of video sequences. Current research has begun to investigate the recognition of human activities taking place in the scene. Applications such as video database, virtual reality interfaces, and smart surveillance systems all have in common tracking and interpreting human activities. Indoor surveillance provides information about areas such as building lobbies, hallways, and offices. Monitoring in lobbies and hallways include detection of people depositing things (unattended luggage in an airport lounge), removing things (theft) or loitering. Outdoor surveillance includes tasks such as monitoring of a site for intrusion or threats from vehicle (e.g., car bombs). In these applications, people are the key element of the system. The ability to detect and track people with their body parts is therefore an important problem.

Our research is motivated by considerations of a ground-base surveillance system that monitors an extended area for human activities. The surveillance system must detect moving objects and identify

them as humans, animals, vehicles. When one or more persons are detected, their movements need to be analyzed to recognize the activities that they are involved in.

We propose a low-cost PC based real-time visual surveillance system, called W^4, for tracking people and their body parts, and monitoring their activities in monochromatic and stereo imagery. It operates on grayscale video imagery, or on video imagery from an infrared camera. Unlike many systems for tracking people, our system make no use of color cue. Instead W^4 employs a combination of shape analysis, robust tracking techniques, and a silhouette based body model to locate and track people, and understand the interactions between people and objects - e.g., people exchanging objects, leaving objects in the scene. A subsequent system, W^4S, integrates real-time stereo computation into W4. Incorporation of stereo has allowed us to overcome the difficulties that W^4 encountered with sudden illumination changes, shadows and occlusion which makes tracking much harder in intensity images. Also we propose a new silhouette-based body model to determine the location of the body parts while people are in generic postures. It is a combination of a hierarchical body pose estimation, a convex hull analysis of the silhouette, and a partial mapping from the body parts to the silhouette segments using a distance transform method.

The proposed system has four main components:

- Modeling the monitored area
- Detecting and identifying people
- Tracking people and their body parts against complex background.
- Understanding interactions between people and objects.

W^4 is a real time system for tracking people and their body parts in monochromatic imagery. It constructs dynamic models of people's movements to answer questions about *what* they are doing, and *where* and *when* they act. It constructs appearance models of the people it tracks so that it can track people (*who?*) through occlusion events in the imagery. We describe the computational models employed by W^4 to detect and track people and their parts. These models are designed to allow W^4 to determine types of interactions between people and objects, and to overcome the inevitable errors and ambiguities that arise in dynamic image analysis (such as instability in segmentation processes over time, splitting of objects due to coincidental alignment of objects parts with similarly colored background regions, etc.) W^4 employs a combination of shape analysis and robust techniques for tracking to detect people, and to locate and track their body parts. It builds "appearance" models of people so that they can be identified after occlusions or after other interactions during which W^4 cannot track them individually.

W^4 has been designed to work with only monochromatic video sources, either visible or infrared. While most previous work on detection and tracking of people has relied heavily on color cues, W^4 is designed for outdoor surveillance tasks, and particularly for night-time or other low light level situations. In such cases, color will not be available, and people need to be detected and tracked based on weaker appearance and motion cues. W^4 is a real time system. It currently is implemented on a dual processor Pentium PC and can process between 10-30 frames per second depending on the image resolution (typically lower for IR sensors than video sensors) and the number of people in its field of view.

W^4 currently operates on video taken from a stationary camera and many of its image analysis algorithms would not generalize easily to images taken from a moving camera. Other ongoing research in our laboratory attempts to develop both appearance and motion cues from a moving sensor that might alert a system to the presence of people in its field of regard [15]. At this point, the surveillance system might stop and invoke a system like W^4 to verify the presence of people and recognize their actions.

Figure 1: Examples of detection result: intensity image (left), detected people and their body parts form the intensity only (middle) and their placement in the $2\frac{1}{2}D$ scene by W^4S (right)

W^4S is a subsequent version of W^4 which represents the integration of a real-time stereo (SVM) system into W^4 to increase its reliability. SVM [26] is a compact, inexpensive realtime device for computing dense stereo range images which was recently developed by SRI. The incorporation of stereo has allowed us to overcome the difficulties that W^4 encountered with sudden illumination changes, shadows and occlusions. Even low resolution range maps allow us to continue to track people successfully, since stereo analysis is not significantly effected by sudden illumination changes and shadows, which make tracking much harder in intensity images. Stereo is also very helpful in analyzing occlusions and other interactions. W^4S has the capability to construct a $2\frac{1}{2}D$ model of the scene and its human inhabitants by combining a 2D cardboard model, which represents the relative positions and size of the body parts, and range as shown in figure 1.

W^4 will be extended with models to recognize the actions of the people it tracks. Specifically, we are interested in interactions between people and objects- e.g., people exchanging objects, leaving objects in the scene, taking objects from the scene. The description of the people-their global motion and the motion of the their body parts- developed by W^4 are designed to support such activity recognition.

A new silhouette based human model is proposed to perform body part analysis. *Ghost* is a real time system for detecting human body parts in monochromatic imagery. It constructs a silhouette-based body model to determine the location of the body parts which are used to recognize human activities in the surveillance systems. We describe the computational models employed by *Ghost* to predict the location of the six main body parts (e.g, the head, hands(2), feet(2), and the torso) while the person is in any posture. *Ghost* works under the control of W^4. W^4 invokes *Ghost* when it needs to locate the body parts (when W^4 detects a new person or when it loses a body part which was being tracked); *Ghost* locates the body parts and passes their estimated locations to W^4.

2 Related Work

In this section, we summarize some of the related work. We first summarize the real-time people tracking system, activity recognition research which could be applicable to our system, and human body part labeling algorithms.

2.1 People Tracking System

Olson [32] described a set of algorithms for the core computation needed to build smart cameras. They build a system, called AVS, which is a general-purpose framework for moving object detection and event recognition. Moving objects are detected using change detection, and tracked using first-order

prediction and nearest neighbor matching. Events are recognized by applying predicates to the graph formed by linking corresponding objects in successive frames.

Pfinder [48] has evolved several years and has been used to recover a 3D description of a person in a large room size space. Pfinder has been used in many applications. It solves the problem of person tracking in arbitrary complex scenes in which there is a single unoccluded person and fixed camera. Pfinder utilizes a 2D image analysis architecture with two complementary procedures for 2D tracking and initialization. The initialization procedure obtains descriptions of the person from weak prior knowledge about the scene and is necessary for bootstrapping the system at the start and when tracking break downs. The tracking procedure recursively updates the description based on strong priors from the previous frame. The tracking procedure can determine when it is in error and can then defer to initialization procedure, which is slower but more reliable. Initialization and tracking procedures for pfinder is based largely on a Maximum A Posteriori probability (MAP) approach. Spfinder[3] is a recent extension of Pfinder in which a wide-baseline stereo camera is used to obtain 3-D models. Spfinder has been used in a smaller desk-area environment to capture accurate 3D movements of head and hands. Kanade [25] has implemented a Z-keying method, where a subject is placed in correct alignment with a virtual world.

KidRooms [6, 20, 21] is a tracking system based on "closed-world regions". These are regions of space and time in which the specific context of what is in the regions is assumed to be known. These regions are tracked in real-time domains where object motions are not smooth or rigid, and where multiple objects are interacting. Multivariate Gaussian models are applied to find the most likely match of human subjects between consecutive frames taken by cameras mounted in various location in [10]. Bregler uses many levels of representation based on mixture models, EM, and recursive Kalman and Markov estimation to learn and recognize human dynamics [9].

2.2 Activity Recognition

Activities involve a sequence of motions. The components of the sequence can either be movements or static states. The representation of the sequence defining an activity can either be explicit and deterministic, or implicit and statistical. In [19], the positions of the silhouette edges of a walking person are encoded as as a one degree of freedom function of the phase of the gait. These edges are matched against those of a person in each frame. The gait phase is then estimated at each time yielding a description of the complete sequence in terms of trajectory through the gait phases.

Applications of Hidden Markov Models to understanding human gesture are the statistical representation of the sequences. Hidden Markov Models represent activities by probabilistic states where both the observed output of a given state and the transition made between states are controlled by underlying probability distributions. The use of HMMs to encode the statistical sequence of movements or states associated with a activity has both advantages and disadvantages. The most important advantage of HMMs is their ability to learn the necessary states and transitions from training examples. The learning algorithm decompose the activity into natural phases. However, the disadvantages of HMMs is the lack of control one has over the states recovered. It is difficult to incorporate priori knowledge about activity segments.

In Starner's work [44], they describe an extensible system which uses a single color camera to track a hand in real time and interprets American Sign Language (ASL) using Hidden Markov Models. The tracking process produces only a coarse description of hand shape, orientation, and trajectory. Currently, in their system, a person is required to wear colored gloves to facilitate the hand tracking. This shape, orientation, and trajectory information is then input to a HMM for recognition of the signed word. Through use of Hidden Markov Models, low error rates were achieved on both the training set and

independent test set without invoking a complex model of hands.

Yamato [51] proposed a new human action recognition method based on a Hidden Markov Model. They used a feature based, bottom up approach with HMMs that is characterized by its learning capability and time-scale invariability. To apply HMMs to their system, one set of time-sequential images is transformed into an image feature vector sequence, and the sequence is converted into a symbol sequence by vector quantization. In learning human action categories, the parameters of the HMMs, one per category, are optimized so as to best describe the training sequences from the category. To recognize the observed sequence, the HMM which best matches the sequence is chosen. They applied the method to real-time sports action (tennis players) and achieved 90% recognition rates.

Brand [8] presented an algorithm for coupling and training Hidden Markov Models to model interaction processes, and demonstrated their superiority to conventional HMMs in a vision task classifying two-handed actions (Tai Chi Chu actions). He claimed that a conventional HMM is the wrong model in that most interesting signals fail to satisfy the very restrictive Markov Model, e.g, multiple interacting processes that have structure in both time and space. For example, in a video signal one might want to model the behavior of players in a sport or participants in multi-place action verbs such as "A gave B the C". He presented algorithm for coupling and training HMMs to model interactions between processes that may have different structures and degree of causal (temporal) influence on each other. Coupled Hidden Markov Models offers a way to model multiple interacting. processes Noncausal HMM coupling can represent the fact that it is unlikely to see both players playing net simultaneously; the causal HMM coupling can represent the fact that one player rushing to the net will drive the other back and restrict the kind of returns he attempts.

The recognition of activities usually requires (1) a statistical or deterministic representation of sequence of activity segments, 2) a parsing mechanism that can temporarily align the input signal with known activity patterns.

An alternative to HMMs for recovery to natural gestures was proposed by Wilson. Wilson [47] presented a method for recovery of the temporal structure (the gestural phases) and phases in natural gesture. He studied four types of gesture generated during discourse, iconic where the motion or configuration of hands physically match the object or situation of narration; deictic: a point gesture; metaphoric, where the motion or shape of the hands is somehow suggestive of the situation; beats which generated to show emphasis or to repair mis-spoken segments. He characterized these gesture types as temporal signatures into two group. "Bi-phasic" gestures which consists of a small baton-like movement away from th rest state and then back again and "tri-phasic" gestures which are executed by first transitioning from the rest phase into gesture space, then executing small movement, remaining there a short duration, and then transitioning back to rest state. In his work, he tried to find possible instances of bi- nd tri-phasic gestures in a video sequence of someone telling a story. He used eigenvector decomposition of the images to represent the images (10 eigen cofficient). He applied the technique to image sequences by randomly selecting a few hundred frames, computing eigen vector decomposition of these frames, and then projecting all frames onto the resulting basis. A Markovian state was used description to detect gesture phase by detecting the rest states first.

Another approach is principle components analysis (PCA) for activity recognition. Yacoob [50] presented a framework for modeling and recognition of temporal activities by modeling of sets of exemplar activities and parameterizing their representations in the form of principal components. Recognition of spatio-temporal variants of modeled activities is achieved by parameterizing the search in the space of admissible transformations that the activities can undergo. Experiments on recognition of articulated and deformable object motion from image motion parameters were presented. An observed activity can be viewed as a vector of measurements over the temporal axis. In their paper, they showed that a

reduced dimensionality model of activities such as "walking" can be constructed using principle component analysis (PCA) of example signals. Recognition of such activities is then posed as matching between principal component representation of the observed activity to these learned models that may be subjected to "activity-preserving" transformations.

Dynamic time warping was also applied to activity recognition. Darrel [13] proposed a method for learning, tracking and recognizing human gestures using a view-based approach to model both object and behavior. Object view are represented using sets of view models, rather than single templates. Stereotypical space-time pattern, gestures are then matched to stored gesture patterns using dynamic time warping. Real-time performance was archived by using special-purpose correlation hardware and view prediction to prune as much of the search space possible.

Polana [38, 40, 39] used motion information in activity recognition. They demonstrated a general computational method to recognize such movements in real image sequences using what is essentially template matching in a motion feature space coupled with a technique for detecting and normalizing periodic activities.

2.3 Human Body Part Labeling

Iwasawa[22] introduces a new real-time method to estimate the posture of a human from thermal images acquired by an infrared camera regardless of the background and lighting conditions. Distance transformation is performed for the human body area extracted from the thresholded thermal image for the calculation of the center of gravity.

Leung and Yang [29, 30] tried to solve the body labeling problem in a sequence of images. They describe a segmentation method, using difference images and past history. The advantage of difference images is that motion can be localize to a particular portion of the image. History referencing handles the case where no current motion is detected for a region, but where motion was detected, for that same region, in the previous frame. The process is made up of two steps: region description and body part identification. The region description process extracts anti parallel lines using segmented regions and then a growing procedure which recursively finds a new pairs of anti parallel lines and concatenates them to the upper part. After selection of anti parallel lines, a labeling algorithm is applied. Labeling is a mapping to anti parallel lines to model by using some experimentally determined constraints on the length/width ratio of body parts with respect to whole body length/width.

Akita [1] used a key frame sequence of stick figures to approximately predict the location of body parts with respect to other part. The actual body is modeled using generalized cones. However, only their projection on the image plane is used. The recognition of the parts is done in following order: legs, head, arms and trunk. The determination of legs, head, arms and trunk parameters is performed at every frame. Correspondence between frames is established using one of two methods. When the positions change of a segment is small enough, its position can be predicted from previous frame using window code distance. If window codes can not find a correspondence, then a key frame sequence is used to find the current posture of the body. The position of the body parts is then recomputed.

Kono [27] described a robust and reliable method of human detection for visual surveillance system. They first take precise silhouette patterns by detecting and analyzing the change in the brightness between the background image and the current image. They used the shape features of the silhouette patters of humans as the detection parameters. The shape feature are mainly the mean and the standard deviation of the projection histogram of the silhouette pattern. The basic values of the shape features are decided from 200 typical silhouette patterns of walking. If the difference between the basic values and the values calculated from a silhouette pattern is small, the silhouette pattern is judged as a human.

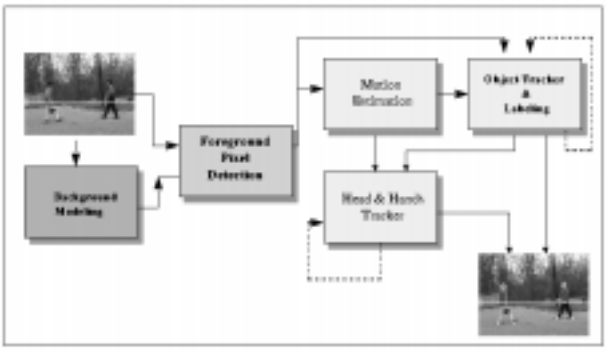


Figure 2: Detection and Tracking System

3 Prior Research - W^4: A Real Time System for Detecting and Tracking People

W^4 is a real time system for tracking people and their body parts in monochromatic imagery. It constructs dynamic models of people's movements to answer questions about what they are doing, and where and when they act. It constructs appearance models of the people it tracks so that it can track people (who?) through occlusion events in the imagery. We describe the computational models employed by W^4 to detect and track people and their parts. These models are designed to allow W^4 to determine types of interactions between people and objects, and to overcome the inevitable errors and ambiguities that arise in dynamic image analysis (such as instability in segmentation processes over time, splitting of objects due to coincidental alignment of objects parts with similarly colored background regions, etc.) W^4 employs a combination of shape analysis and robust techniques for tracking to detect people, and to locate and track their body parts. It builds "appearance" models of people so that they can be identified after occlusions or after other interactions during which W^4 cannot track them individually.

W^4 has been designed to work with only monochromatic video sources, either visible or infrared. While most previous work on detection and tracking of people has relied heavily on color cues, W^4 is designed for outdoor surveillance tasks, and particularly for night-time or other low light level situations. In such cases, color will not be available, and people need to be detected and tracked based on weaker appearance and motion cues. W^4 is a real time system. It currently is implemented on a dual processor Pentium PC and can process between 10-30 frames per second depending on the image resolution (typically lower for IR sensors than video sensors) and the number of people in its field of view.

The system diagram of W^4 is shown in Figure 2. In W^4, foreground regions are detected in every frame by a combination of background analysis and simple low level processing of the resulting binary image. The background scene is statically modeled by the minimum and maximum intensity values and maximal temporal derivative for each pixel recorded over some period, and is updated periodically. Each foreground region is matched to the current set of objects using a combination of shape analysis and tracking. These include simple spatial occupancy overlap tests between the predicted locations of objects and the locations of detected foreground regions, and "dynamic" template matching algorithms that correlate evolving appearance models of objects with foreground regions. Second-order motion models, which combine robust techniques for region tracking and matching of silhouette edges with recursive least square estimation, are used to predict the locations of objects in future frames. A

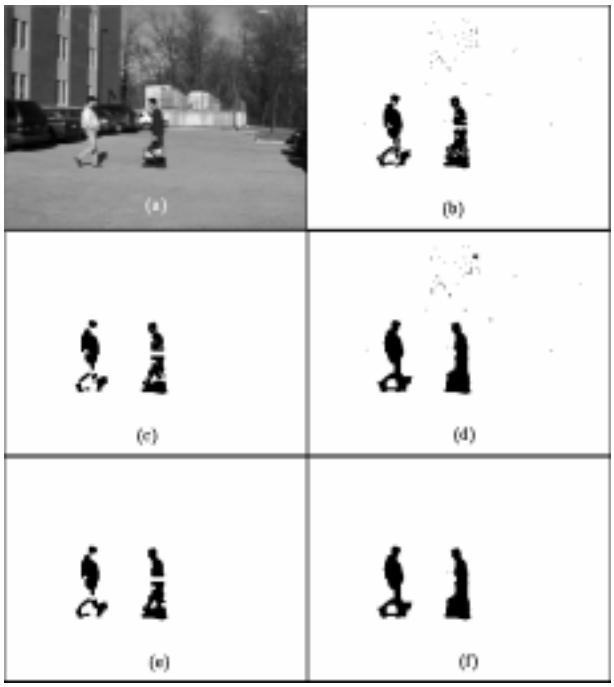


Figure 3: Foreground pixels detected by using different methods;thresholding only (b), one iteration of erosion and one iteration of dilation (c), two iterations of dilation and two iteration erosion (d), one iteration of erosion, two iterations of dilation and one iteration of erosion (e), and the result of W^4(f)

cardboard human model of a person in a standard upright pose is used to model the human body and to predict the location of human body parts (head, torso, hands, legs and feet). The locations of these parts are verified and refined using dynamic template matching methods. W^4 can detect and track multiple people in complicated scenes at 20 Hz speed for 320x240 resolution on 200 MHz dual pentium PC. W^4 has also been applied to infrared video imagery at 30Hz for 160x120 resolution on the same PC.

3.1 Background Scene Modeling and Foreground Region Detection

Frame differencing in W^4 is based on a model of background variation obtained while the scene contains no people. The background scene is modeled by representing each pixel by three values; its minimum and maximum intensity values and the maximum intensity difference between consecutive frames observed during this training period. These values are estimated over several seconds of video and are updated periodically for those parts of the scene that W^4 determines to contain no foreground objects. The mean and standard deviation of each pixel have been employed in other system [48, 32] for background scene modeling. Our experiments suggested that these methods are not as reliable for distinguishing between real foreground objects and the small motions of background objects in outdoor video imagery. For example, the small motions of leaves of a tree should be belong to background; they should not detected as foreground objects.

Foreground objects are segmented from the background in each frame of the video sequence by a four stage process: thresholding, noise cleaning, morphological filtering and object detection.

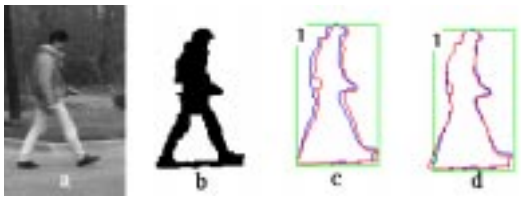


Figure 4: Motion estimation of body using Silhouette Edge Matching between two successive frame a: input image; b: detected foreground regions; c: alignment of silhouette edges based on difference in median; d: final alignment after silhouette correlation

Figure 5: An example of true-split interaction: a person brings an object into the scene and places it on the ground. W^4 detected that the small object was introduced by object 1

Each pixel is first classified as either a background or a foreground pixel using the background model. Giving the minimum (M), maximum (N) and the largest interframe absolute difference (D) images that represent the background scene model, pixel x from image I is a foreground pixel if:

$$|M(x) - I(x)| > D(x) \quad \text{or} \quad |N(x) - I(x)| > D(x) \tag{1}$$

Thresholding alone, however, is not sufficient to obtain clear foreground regions; it results in a significant level of noise, for example, due to illumination changes. W^4 uses region-based noise cleaning to eliminate noise regions. After thresholding, one iteration of erosion is applied to foreground pixels to eliminate one-pixel thick noise. Then, a fast binary connected-component operator is applied to find the foreground regions, and small regions are eliminated. Since the remaining regions are smaller than the original ones, they should be restored to their original sizes by processes such as erosion and dilation.

Generally, finding a satisfactory combination of erosion and dilation steps is quite difficult, and no fixed combination works well, in general on our outdoor images. Instead, W^4 applies morphological operators to foreground pixels only after noise pixels are eliminated. So, W^4 reapplies background subtraction, followed by one iteration each of dilation and erosion, but only to those pixels inside the bounding boxes of the foreground regions that survived the size thresholding operation.

As the final step of foreground region detection, a binary connected component analysis is applied to the foreground pixels to assign a unique label to each foreground object. W^4 generates a set of features for each detected foreground object, including its local label, centroid, median, and bounding box.

3.2 Object Tracking

The goals of the object tracking stage are to:

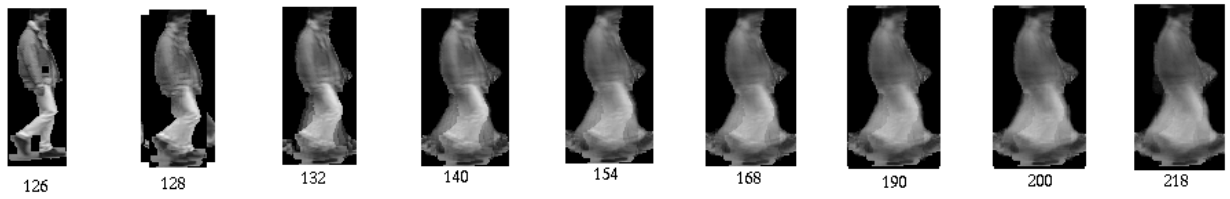


Figure 6: An example of how temporal templates are updated over time

- determine when a new object enters the system's field of view, and initialize motion models for tracking that object.
- compute the correspondence between the foreground regions detected by the background subtraction and the objects currently being tracked by W^4.
- employ tracking algorithms to estimate the position (of the torso) of each object, and update the motion model used for tracking. W^4 employs second order motion models (including a velocity and, possibly zero, acceleration terms) to model both the overall motion of a person and the motions of its parts.

W^4 has to continue to track objects even in the event that its low level detection algorithms fail to segment people as single foreground objects. This might occur because an object becomes temporarily occluded (by some fixed object in the scene), or an object splits into pieces (possibly due to a person depositing an object in the scene, or a person being partially occluded by a small object). Finally, separately tracked objects might merge into one because of interactions between people. Under these conditions, the global shape analysis and tracking algorithms generally employed by W^4 will fail, and the system, instead, relies on local correlation techniques to attempt to track parts of the interacting objects.

W^4 first matches objects to current foreground regions by finding overlap between the estimated (via the global motion model) bounding boxes of objects and the bounding boxes of foreground regions from the current frame. For each object, all current foreground regions whose bounding boxes overlap sufficiently are candidates for matching that object. Ideally, one to one matching (tracking) would be found while tracking one object. However, one to many (one tracked object splits into several foreground regions), many to one (two or more tracked objects merge into one foreground region), one to zero (disappearing) and zero to one (appearing) matchings occur frequently. W^4 tracks objects using different methods under each condition.

3.2.1 Appearing Objects

When a foreground region is detected whose bounding box does not sufficiently overlap any of the existing objects, it is not immediately evident whether it is a true object or a noise region. If the region can be tracked successfully through several frames, then it is added to the list of objects to be tracked.

3.2.2 Tracking

Here, we consider the situation that an object continues to be tracked as a single foreground region. W^4 employs a second order motion model for each object to estimate its location in subsequent frames. The prediction from this model is used to estimate a bounding box location for each object. These predicted bounding boxes are then compared to the actual bounding boxes of the detected foreground regions. Given that an object is matched to a single foreground region (and the sizes of those regions are roughly the same) W^4 has to determine the current position of the object to update its motion model. Even though the total motion of an object is relatively small between frames, the large changes in shape of the silhouette of a person in motion causes simple techniques, such as tracking the centroids of the foreground regions, to fail. Instead, W^4 uses a two stage matching strategy to update its global position estimate of an object. The initial estimate of object displacement is computed as the motion of the **median** coordinate of the object. This median coordinate is a more robust estimate of object position, and is not effected by the large motions of the extremities (which tend to influence the centroid significantly). It allows us to quickly narrow the search space for the motion of the object. However, this estimate is not accurate enough for long term tracking. Therefore, after displacing the silhouette of the object from the previous frame by the median-based estimate, we perform a binary edge correlation between the current and previous silhouette edge profiles. This correlation is computed only over a 5x3 set of displacements. Typically, the correlation is dominated by the torso and head edges, whose shape changes slowly from frame to frame. This tracking process is illustrated in figure 4.

3.2.3 Region splitting

An object being tracked might split into several foreground regions, either due to partial occlusion or because a person deposits an object into the scene. An example of the latter is illustrated in figure 5. In this case, one object will be matched to two or more current foreground regions. W^4 determines whether the split is a true-split or a false-split (due to noise transient) condition by monitoring subsequent frames, while tracking the split objects as individual objects. If W^4 can track the constituent objects over several frames, then it assumes that they are separate objects and begins to track them individually.

3.2.4 Region merging

When two people meet they are segmented as one foreground region by the background subtraction algorithm. W^4 recognizes that this occurs based on a simple analysis of the predicted bounding boxes of the tracked objects and the bounding box of the detected (merged) foreground region. The merged region is tracked until it splits back into its constituent objects. Since the silhouette of the merged regions tends to change shape quickly and unpredictably, W^4 uses a simple extrapolation method to construct its predictive motion model for the merged objects.

A problem that arises when the merged region splits, and the people "re-appear", is determining the correspondence between the people that were tracked before the interaction and the people that emerge from the interaction. To accomplish this, W^4 uses two types of appearance models that it constructs while it is tracking an isolated person.

W^4 constructs a dynamic template -called a *temporal texture template* - while it is tracking an isolated object. The temporal texture template for an object is defined by:

$$\Psi^t(x,y) = \frac{I(x,y) + w^{t-1}(x,y) \times \Psi^{t-1}(x,y)}{w^{t-1}(x,y) + 1} \tag{2}$$

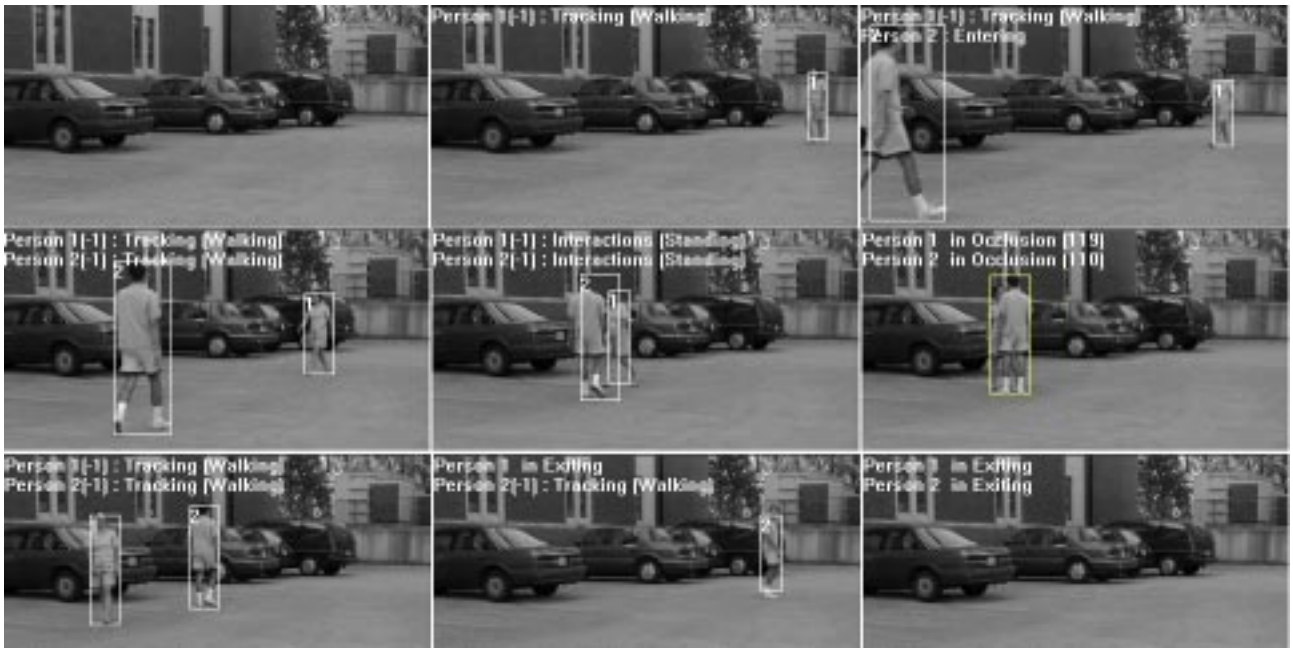


Figure 7: An example for W^4 tracking; two people are entering, walking, meeting and leaving

Here, I refers to the foreground region detected during tracking of the object, and all coordinates are represented relative to the **median** of the template or foreground region. The weights in (2) are the frequency that a pixel in Ψ is detected as a foreground pixel during tracking. The initial weights $w^t(x, y)$ of Ψ are zero and are incremented each time that the corresponding location (relative to the median template coordinate)is detected as a foreground pixel in the input image. An example of how the temporal texture template of a person evolves over time is shown in figure 6.

After separation, each constituent object is matched with the separating objects by correlating their temporal templates. Since the temporal texture template is view-based, it could fail to match if there were a large change in the pose of the object during the occlusion event. Therefore, a non-view-based method, which uses a symbolic object representation, is also used to analyze the occlusion. For example, if the temporal texture templates fail to yield sufficiently high correlation values, then we match objects based on the average intensities in their upper, lower and middle parts, in an attempt to identify objects when they separates.

Figure 7 illustrates W^4 tracking objects; W^4 detects people, assigns unique labels to them and tracks them through occlusion and interaction.

3.3 Tracking People's Parts

In addition to tracking the body as a whole, we want to locate body parts such as the head, hands, torso, legs and feet, and track them in order to understand actions. W^4 uses a combination of shape analysis and template matching to track these parts (when a person is occluded, and its shape is not easily predictable, then only template matching is used to track body parts). The shape model is implemented using a a *Cardboard Model* [43] which represents the relative positions and sizes of the body parts. Along with second order predictive motion models of the body and its parts, the Cardboard

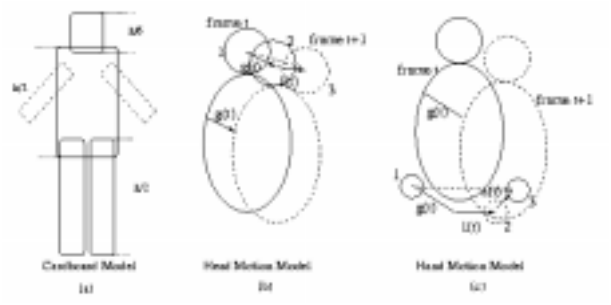


Figure 8: Cardboard model used in W^4 (a), and motion models used for the head (b) and hands (c).

Model can be used to predict the positions of the individual body parts from frame to frame. Figure 8 illustrates the motion models used for the hands and head. These positions are verified (and refined) using dynamic template matching based on the temporal texture templates of the observed body parts.

The cardboard model represents a person who is in an upright standing pose, as shown in figure 8(a). It is used to predict the locations of the body parts (head, torso, feet, hands, legs). The height of the bounding box of an object is taken as a height of the cardboard model. Then, fixed vertical scales are used to determine the initial approximate location (bounding box) of individual body parts, as shown in Figure 9. The lengths of the initial bounding boxes of the head, torso, and legs are calculated as 1/5, 1/2 and 1/2 of the length of bounding box of the object, respectively. The widths of the bounding boxes of the head, torso, and legs are calculated by finding the median width (horizontal line widths) inside their initial bounding boxes. In addition to finding sizes and locations, the moments of the foreground pixels inside the initial bounding boxes are calculated for estimating their principal axis. The principal axis provide information about the pose of the parts. The head is located first, followed by the torso and legs. The hands are located after the torso by finding extreme regions which are connected to the torso and are outside of the torso. The feet are located as extreme regions in the direction of the principal axes of the respective leg. Figure 9 show an example of how the cardboard model can be used to predict the locations of body parts in two stages (approximate initial location and final estimated location) and represent them as ellipsis.

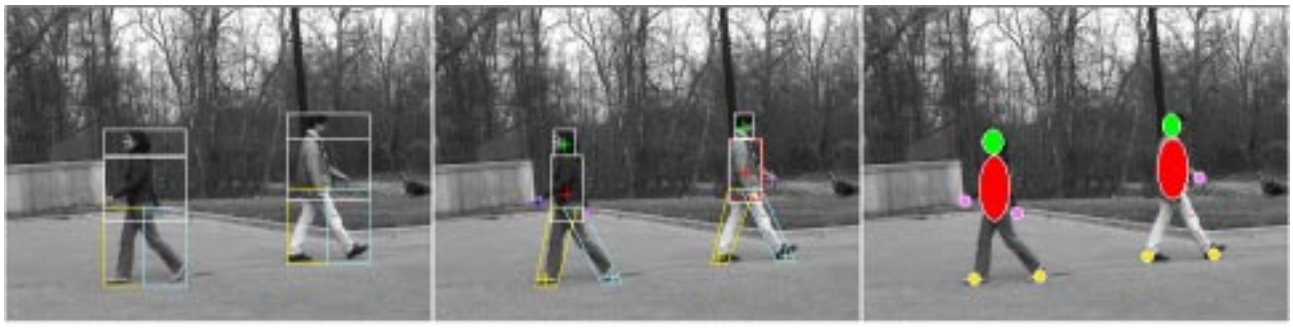


Figure 9: An example of Cardboard Model to show How Head, Torso, Legs Hands and Feet are Located. Initial Bounding boxes are located on foreground regions (a); Cardboard model analysis locates the body part (b); illustration of body part location by ellipsis (c)

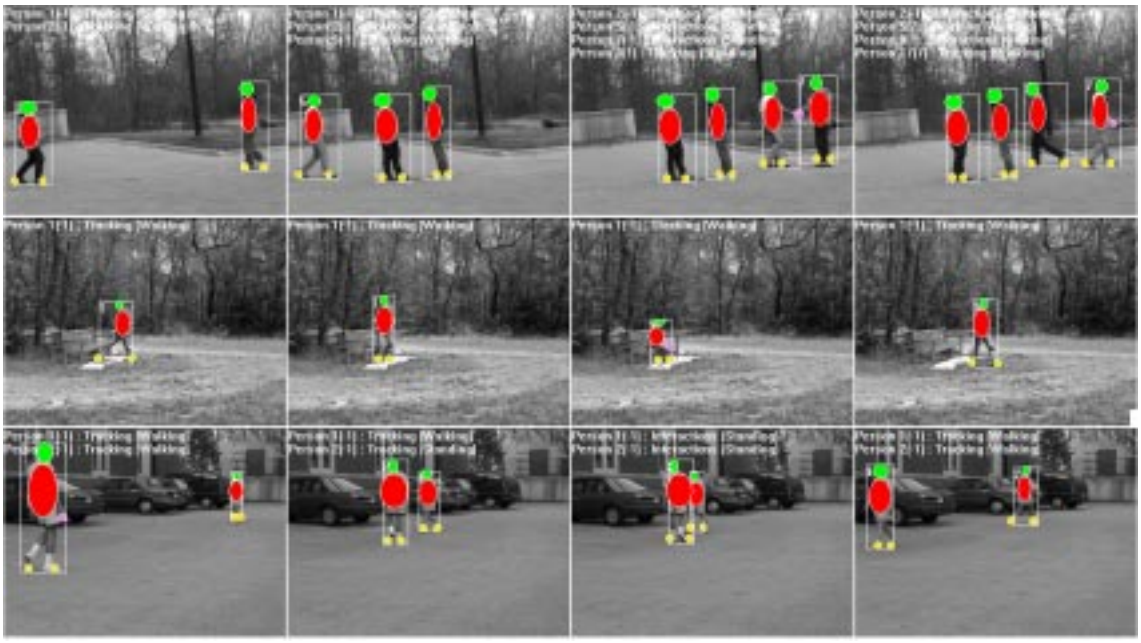


Figure 10: Examples of using the cardboard model to locate the body parts in different actions: four people meet and talk (first line), a person sits on a bench (second line), two people meet (third line)

After predicting the locations of the head and hands using the cardboard model, their positions are verified and refined using temporal texture templates. These temporal texture templates are then updated as described previously, unless they are located within the silhouette of the torso. In this case, the pixels corresponding to the head and hand are embedded in the larger component corresponding to the torso. This makes it difficult to accurately estimate the median position of the part, or to determine which pixels within the torso are actual part pixels. In these cases, the parts are tracked using correlation, but the templates are not updated.

Figure 10 and Figure 11 illustrates some results of W^4 system in scenes of parking lots and parkland.

The correlation results are monitored during tracking to determine if the correlation is good enough to track the parts correctly. Analyzing the changes in the correlation scores allows us to make predictions about whether a part is becoming occluded. For example, the graph in the Figure 12 shows how the correlation (sum of absolute differences, SAD) results for the left hand of a person changes over time. Time intervals I,II,III and IV in the graph are the frame intervals in which the left hand is occluded by the body, and they have significantly worse correlation scores (higher SAD). and hands are generated and updated. When the head and hands are inside the silhouette of the body, then the current analysis cannot locate them. Therefore, W^4 does not attempt to update head and hand templates; rather, it predicts their locations and tracks them only by correlation.

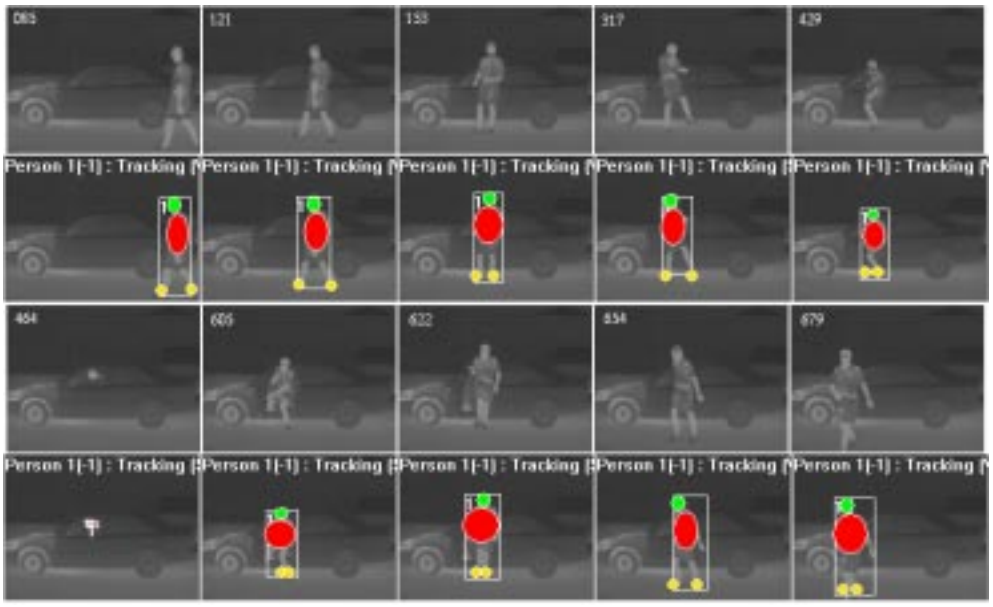


Figure 11: Examples of using the cardboard model to locate the body parts in Infrared Imagery

Figure 12: An Example of hands and head tracking of a walking person and correlation results for left hand of that person.

4 Prior Research - W^4S: Detecting and Tracking People in $2\frac{1}{2}D$

W^4S is a real time system for tracking people and their body parts in stereo imagery. W^4S represents the integration of a real-time stereo (SVM) system with W^4 to increase its reliability. SVM [26] is a compact, inexpensive real time device for computing dense stereo range images which was recently developed by SRI. W^4S employs a combination of shape analysis and robust techniques for tracking to detect people, and to locate and track their body parts using both intensity and stereo. The incorporation of stereo has allowed us to overcome the difficulties that W^4 encountered with sudden illumination changes, shadows and occlusions. Even low resolution range maps allow us to continue to track people successfully, since stereo analysis is not significantly effected by sudden illumination changes and shadows, which make tracking much harder in intensity images. Stereo is also very helpful in analyzing occlusions and other interactions. W^4S has the capability to construct a $2\frac{1}{2}D$ model of the scene and its human inhabitants by combining a 2D cardboard model, which represents the relative positions and size of the body parts,

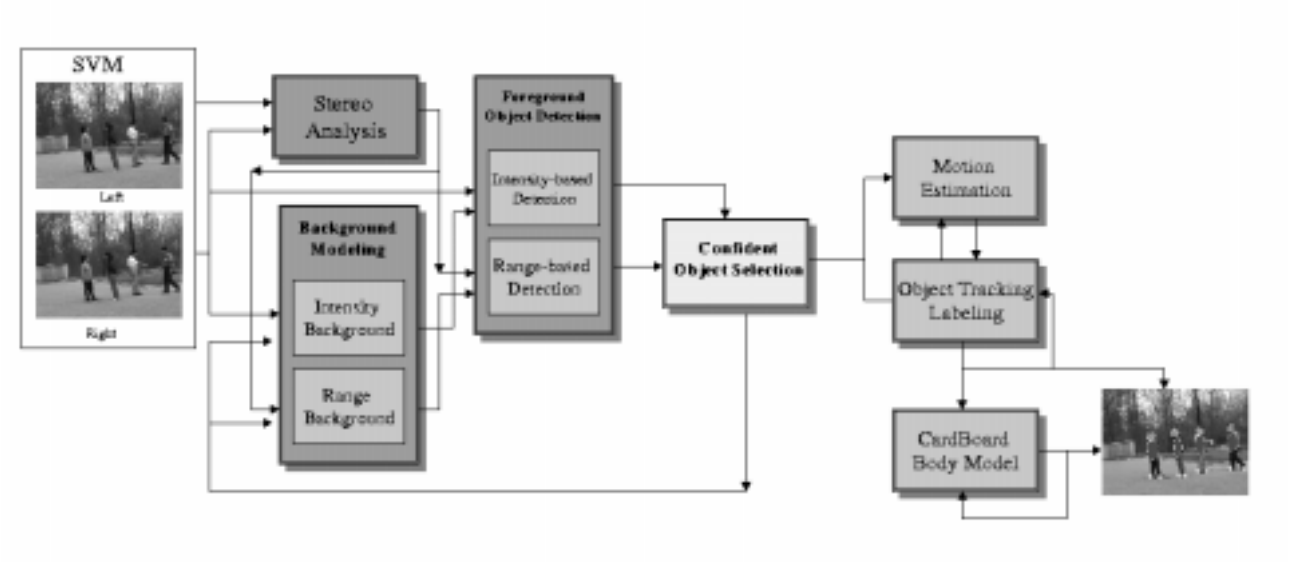


Figure 13: Detection and Tracking System

and range as shown in figure 1.

The system diagram of W^4S is shown in Figure 13. Area-correlation based stereo is computed by SVM. Foreground regions in both the intensity image and the range image are detected in every frame using a combination of background analysis and simple low level processing of the resulting binary images. The background scene is modeled in the same way for the disparity and intensity images. The background scene for the disparity image is statistically modeled by the minimum and maximum disparity value and maximal temporal disparity derivative for each pixel recorded over some period. A similar statistical model is used to model the background scene for the intensity images using intensity values instead of disparity values. Both background models are updated periodically. The foreground regions detected in the intensity and range images are combined into a unified set of foreground regions. Each foreground region is matched to the current set of objects using a combination of shape analysis and tracking. These include simple spatial occupancy overlap tests between the predicted locations of objects and the locations of detected foreground regions, and "dynamic" template matching algorithms that correlate evolving appearance models of objects with foreground regions. Second-order motion models, which combine robust techniques for region tracking and matching of silhouette edges with recursive least square estimation, are used to predict the locations of objects in future frames. W^4S can detect and track multiple people in complicated scenes at 5-20 Hz speed at 320x120 resolution on a 200 MHz dual pentium PC.

4.1 Stereo Analysis

The range information about the objects in the scene can be measured by the stereo analysis by comparing the object projection on two or more images. We should find corresponding elements to match them between the images to calculate the range using simple stereo image geometry. Generally, two main methods can be used to solve corresponding problem:

- *Feature-based methods* tries to find a match between a small region in one image and the other images using features of that small regions. These features could be some discrete features which are view-point and illumination independent, such as, corners

- *Area-based methods* tries to find a match using small patches among the images using correlation.

Feature-based algorithms can find robust matching as they use view-point independent features, but it could be difficult to find and extract those features in all part of the images and it causes to produce sparse range results. However, area-based algorithm produces dense range result which cover all part of the image. Area-based algorithm compares small patches among images using correlation. The area size is a key factor to produce correct range results, since small areas are more likely to be similar in images with different view-point, larger areas produces more correct range results and increases the signal to noise ratio. Correlation of the images is affected by illumination changes, perspective and imaging differences among the images. There are many methods in the literature which correlates not the raw intensity images,but some transform of the intensities in order to produce more reliable range data. Fua [16] used the normalized intensities method to get good correlation using sum of square difference. Poggio [31], Kanade [24] used Laplacian of Gaussian method with some differences in method that they used, such as, zero-crossing, sum of square difference, sum of absolute difference. Woodfill [52] attempted to deal with the problem of outliers, which causes overwhelming the correlation measure. Among those stereo matching methods, absolute difference of the the others. Faugeras [14] used more than two images in order to increase the signal-to-noise ratio of the matching. Using three or more images could overcome the view-point occlusion where the matching part of an object does not appears in the other image. Dense range images sometimes contain false matches because of the uncertainty in the matches. Especially flat areas, which has insufficient textures, are subject to ambiguous matches. These ambiguous matches should be filtered out. The interest operator has been applied as post filtering to reject the low confidence range data in previous stereo system [7]. Also the other method used in previous stereo systems is the left-right check which helps to understand the consistency of the matches. The left-right check looks for consistency in matching from a fixed left image region to a set of right image regions, and back again from the matched right regions to a set of the left images. It is particularly useful at range discontinuities, where directional matching yield different results.

W^4S computes stereo using area (sum of absolute difference) correlation after a Laplacian of Gaussian transform. The stereo algorithm considers sixteen disparity levels, perform postfiltering with an interest operator, and a left-right consistency check, and finally does $4x$ range interpolation. The stereo computation is done either in the SVM or on the host PC, the latter option providing us access to much better cameras than those used in the SVM. SVM is a hardware and software implementation of area correlation stereo which was implemented and developed by Kurt Konolige at SRI. The hardware consists of two CMOS $320x240$ grayscale imagers and lenses, low-power A/D converter, a digital signal processor and a small flash memory for program storage. A detailed description of the SVM can be found in [26]. SVM performs stereo at two resolutions (160x120 or 320X120) in the current implementation, with speed of up to 8 frames per second. The SVM uses CMOS imagers; these are an order of magnitude noisier and less sensitive than corresponding CCD's. Higher quality cameras can be utilized by W^4S, with the stereo algorithm running on the host PC, to obtain better quality disparity images.

Figure 14 shows a typical disparity image produced by the SVM. Higher disparities (closer objects) are brighter. There are 64 possible levels of disparity and disparity 0 (black areas) are regions where the range data is rejected by the post-processor interest operator due to insufficient texture.

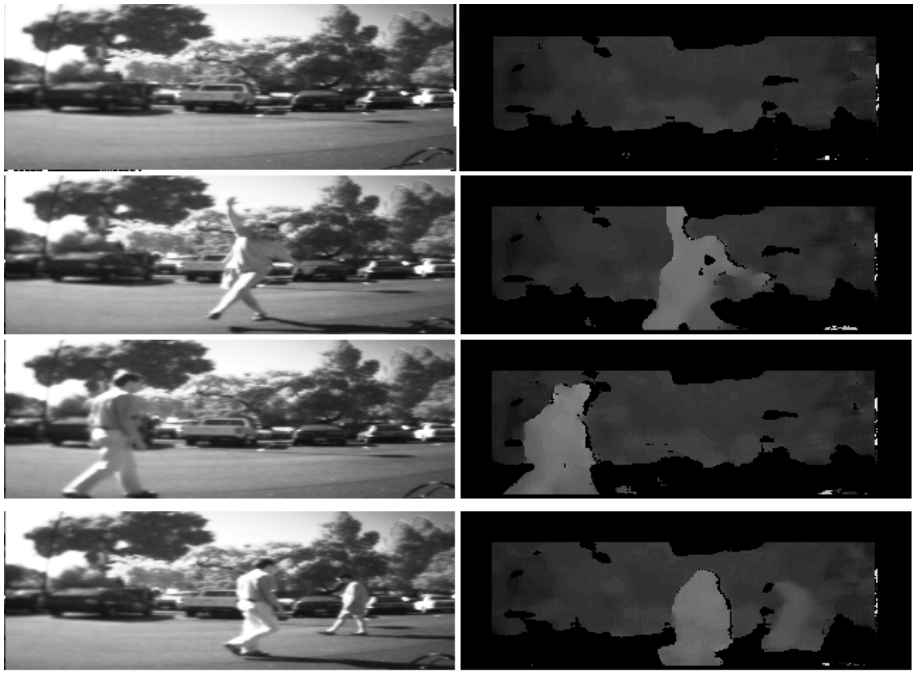


Figure 14: Examples of the range images calculated by area correlation stereo

4.2 Foreground Region Selection

Foreground objects are segmented from the background in each frame of the video sequence by a four stage process: thresholding, noise cleaning, morphological filtering and object detection. This foreground object detection algorithm is simultaneously applied to both the intensity and disparity image. The objects detected in the intensity image and the disparity image are integrated to construct the set of objects that are included in the list of foreground objects, and subsequently tracked. The foreground detection algorithm for intensity images was explained in Section 3.1, the same algorithm is applied to the disparity images in W^4S.

The foreground object detection algorithm is applied to both the disparity images and the intensity images for each frame in the video sequence. Generally, intensity-based detection works better than stereo-based detection when the illumination does not change suddenly or when there are no strong light sources that causes sharp shadows. The main advantage of the intensity-based analysis are that

- Range data may not available in background areas which do not have sufficient texture to measure disparity with high confidence. Changes in those areas will not be detected in the disparity image. However, the intensity-based algorithm can detect low textured areas when the brightness of the foreground regions differs significantly from the background.

- Foreground regions detected by the intensity-based method have more accurate shape (silhouette) then the range-based method. The silhouette is very useful for tracking via motion estimation, and in constructing the appearance-based body model used to locate and track body parts.

However, there are three important situations where the stereo-based detection algorithm has an advantage over the intensity algorithm

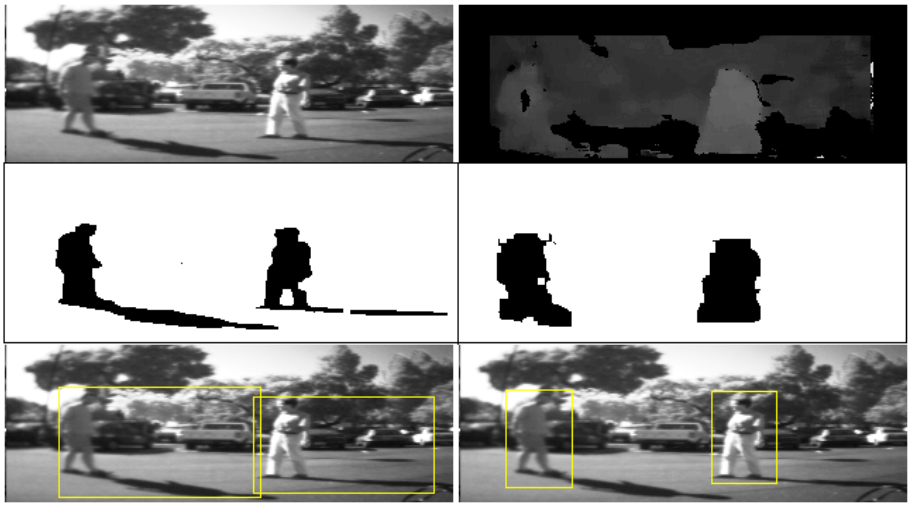


Figure 15: An example showing that how stereo-based detection eliminates shadows during object detection.

- When there is a sudden change in the illumination, it is more likely that the intensity-based detection algorithm will detect background objects as foreground objects. However, the stereo-based detection algorithm is not effected by illumination changes over short periods of time as much as intensity-based detection.
- Shadows, which makes intensity detection and tracking harder, do not cause a problem in the disparity images as disparity does not change from the background model when a shadow is cast on the background. Figure 15 shows the foreground regions and their bounding boxes detected by the intensity based (left) and stereo-based detection (right) methods.
- The stereo-based method more often detects intact foreground objects than the intensity-based method. Intensity-based foreground detection often splits foreground objects into multiple regions due to coincidental alignment of the objects parts with similarly colored background regions.

After foreground objects are detected in both the intensity image and the disparity image, W^4S merges these objects into one set of objects. Objects are selected for tracking as follows:

A graph is constructed in which disparity region is linked to an intensity region that it significantly overlaps ($\geq$60%). The connected components of this graph are then considered as possible foreground objects. Generally, we find large disparity regions with poorly defined shapes that overlap a few intensity regions arising from fragmentation of a foreground object. When a connected component contains only one intensity and one disparity region that have very high overlap, then we represent the foreground object by their intersection.

A connected component is rejected if it contains a region only from the intensity image and disparity is available for that region (but does not differ from the background, since no disparity foreground region was detected). This is probably due to an illumination change or shadow. As the final step of foreground region detection, a binary connected component analysis is applied to the selected foreground regions to assign a unique label to each foreground object. W^4S generates a set of features for each detected foreground object, including its local label, centroid, median, median of the disparity, and bounding box.

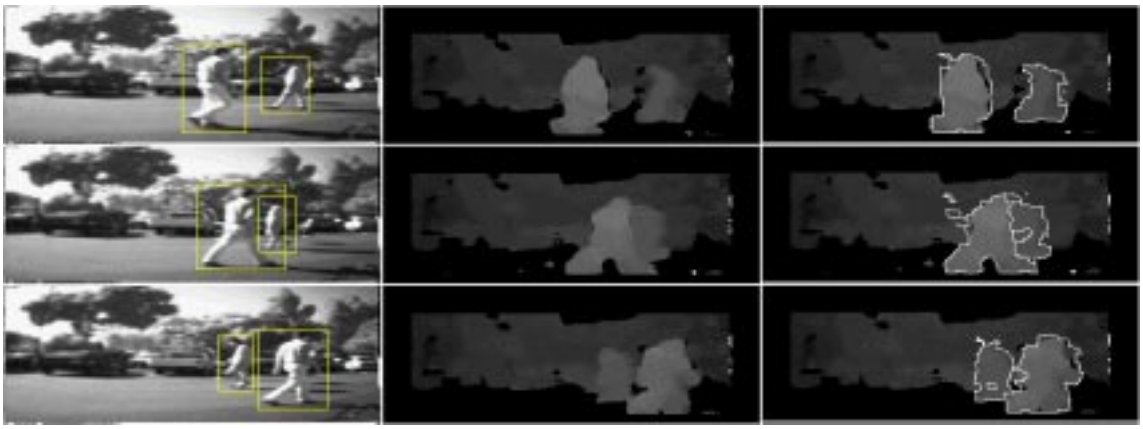


Figure 16: An Example how range data is useful to track the people by segmenting the range data during occlusion

4.3 Object Tracking in $2\frac{1}{2}$

In this section, we'll explain how disparity data is exploited to improve the tracking algorithm used in W^4. The tracking algorithm used in W^4S is very similar to one in W^4 with couple of minor changes to utilize the stereo. Stereo is very helpful in analyzing occlusion and intersection.

W^4S has to continue to track objects even in the event that its low level detection algorithms fail to segment people as single foreground objects. This might occur because an object becomes temporarily occluded (by some fixed object in the scene), or an object splits into pieces (possibly due to a person depositing an object in the scene, or a person being partially occluded by a small object). Finally, separately tracked objects might merge into one because of interactions between people. Under these conditions, the global shape analysis and tracking algorithms generally employed by W^4S will fail, and the system, instead, relies on stereo to locate the objects and local correlation techniques to attempt to track parts of the interacting objects. Stereo is very helpful in analyzing occlusion and intersection. For example, in Figure 16 one can determine which person is closest to the camera when two or more people interact and one is occluded by the other. We can continue to track the people by segmenting the range data into spatially disjoint blobs until the interaction is complete. The range data gives helpful cues to determine "who is who" during and after the occlusion.

When two people meet they are segmented as one foreground region by the background subtraction algorithm. W^4S recognizes that this occurs based on a simple analysis of the predicted bounding boxes of the tracked objects and the bounding box of the detected (merged) foreground region. The merged region is tracked until it splits back into its constituent objects. Since the silhouette of the merged regions tends to change shape quickly and unpredictably, W^4S uses a simple extrapolation method to construct its predictive motion model for the merged objects, and simple disparity segmentation method to predict the location of the objects during interactions. A segmentation is applied to the disparity data only inside the merged regions to locate the interacting objects. A hierarchical connected component algorithm [46] is used for segmentation. The disparity segmentation can successfully distinguish the objects when they are at different ranges as shown in figure 16. Intensity alone cannot determine whether or not the the people are in close proximity when their images merge into one foreground regions. Stereo helps W^4S to recover the relative distance among people when they are in the same foreground region. Figure 17 shows two examples of W^4S tracking people by illustrating their locations

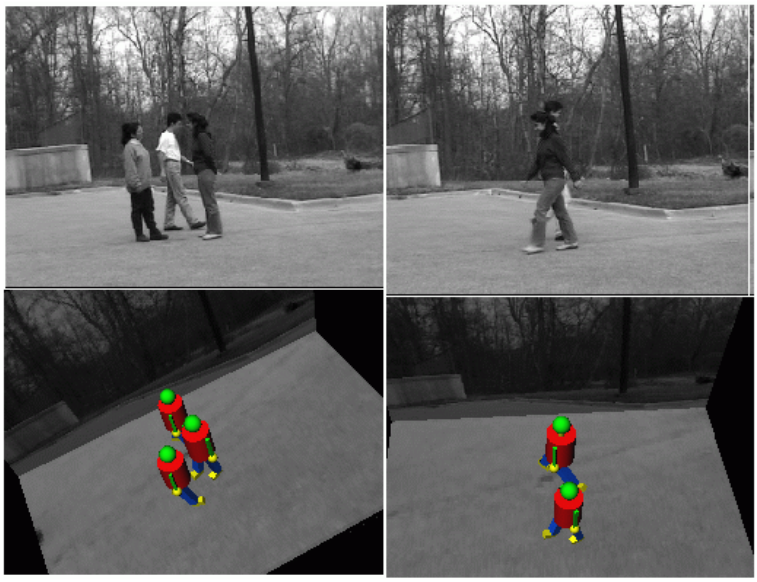


Figure 17: W^4S uses range data to detect the people during occlusion and determine their relative location in $2\frac{1}{2}D$

in $2\frac{1}{2}D$.

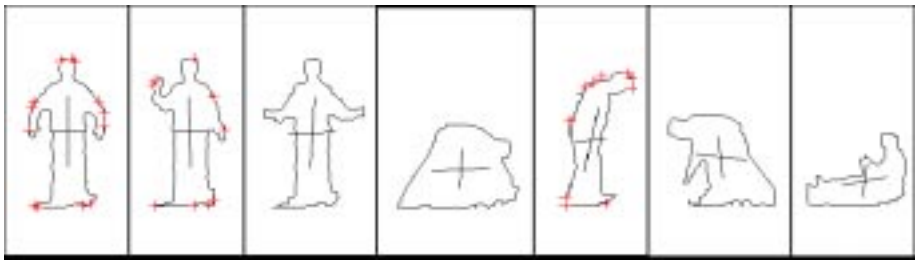


Figure 18: Some silhouette boundaries of the same person in different posture

5 Current Research - *Ghost*: A Human Body Part Labeling System Using Silhouettes

Ghost is a real time system for detecting human body parts in monochromatic imagery. It constructs a silhouette-based body model to determine the location of the body parts which will be used to recognize human activities. In this section, we describe the computational models employed by *Ghost* to predict the location of the six main body parts (e.g, the head, hands(2), feet(2), and the torso) while a person is in any of a number of posture.

W^4 and other previous systems [43, 22, 48, 29, 1] make the assumption that people do actions while they are in an upright-standing posture, and perform their body part analysis based on that assumption. However, a surveillance systems should operate and track body parts while the person is in other postures (e.g., sitting, crawling). We propose an algorithm that works not only in the upright-standing posture but also in other generic postures. A hierarchical posture representation is proposed in *Ghost*. Any body posture is classified into one of four *main postures* (standing, sitting, crawling-bending, and laying down) and then each main posture is classified into one of three *view-based appearance* (front-view, left-side, and right-side).

Ghost works under the control of W^4. W^4 invokes *Ghost* when it needs to locate body parts (when W^4 detects a new person or when it loses a body part which was being tracked). *Ghost* locates the body parts and passes their estimated location to W^4. W^4 then tracks the individual body parts using its correlation method.

Our system is motivated by two basic observations on the relative location of body parts while people are in action.

- It is very likely that the head, hands, elbows, feet, and knees lie on the silhouette boundary.
- The human body in any given posture has a topological structure which constrains the relative locations of body parts. The order of body parts in the silhouette boundary does not change when people do action while they are in the same generic posture (walking), however, the order changes when they change their generic posture (walking to sitting).

Ghost uses a silhouette-based body model which consists of 6 primary body parts (head, hands(2), feet(2), and torso), which we want to locate, and 10 secondary parts(elbows(2), knees(2), shoulders(2), armpits(2), hip, and upper back) which could be on the silhouette boundary and can help to locate the primary parts using the topology of the human body. *Ghost* tries to predict the location of only the primary body parts which are important to human activity recognition. The outline of the algorithm used in *ghost* is as follows:

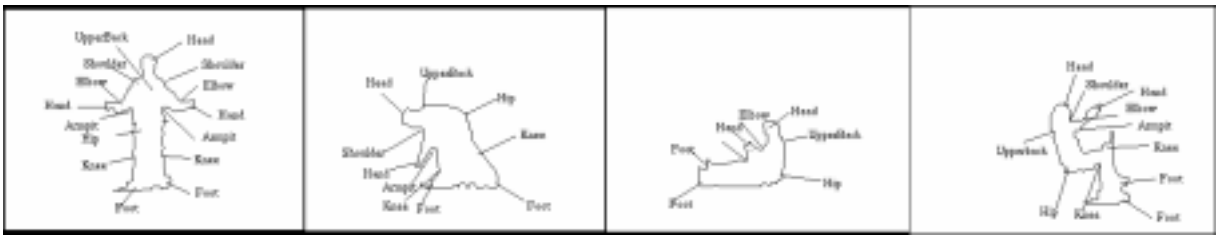


Figure 19: Examples of the order of the body parts on the silhouette boundary

1. A hierarchical body posture analysis is applied to the silhouette to compute the similarities of horizontal and vertical projection histograms of detected silhouette and the main postures. The body posture which yields the highest likelihood in the similarity measure is taken as the estimated posture.

2. A recursive convex-hull algorithm is applied to find possible body part locations on the silhouette boundary.

3. The location of the head is predicted using the major axis of the silhouette, the hull vertices, and the topology of the estimated body posture.

4. When the head location is determined, a topological analysis is applied to eliminate the hull vertices which won't be labeled as a body part, and to map the remaining hull vertices to the body parts using a topological-order preserved distance transform calculation.

5.1 2D body modeling using silhouettes

Ghost uses a silhouette-based body model which consists of 6 primary body parts (head, hands, feet, and torso), which we want to locate, and 10 secondary parts(elbows, knees, shoulders, armpits, hip, and upper back) which could be on the silhouette boundary and can help in locating the primary parts using the topology of the human body (Figure 19). It is not necessary to find all body parts in any frame. Some of them might not be visible from a given point of view.

The primary and secondary body parts should be consistent with the order of the respective main posture (with small variations). These orders are preserved as long as the body stays in the same main posture. For example, if we start from the head (Figure 19) in clockwise order, the main order for the upright-standing pose is head-shoulder-elbow-hand-armpit-knee-foot-foot-knee-armpit-hand-elbow-shoulder-head. This order could vary for different view points. Some parts could be missing on the silhouette boundary or some parts are switched in the order locally (elbow-hand or hand-elbow) because of relative motion of the parts or local occlusions. However, the relative location of some parts (head, feet) should be preserved. For example, head-elbow-shoulder or hand-feet-knee are unacceptable partial orders for the standing posture. Any order of the body parts in the given silhouette should be generated from the order of the main posture by deleting the missing parts or switching the location of some neighbor parts (elbow-hand). Therefore, if we know the posture of the given silhouette and the location of at least one body part, the labeling problem becomes one of mapping the set of body parts to the set of the silhouette segments without violating the expected order of the respective posture.

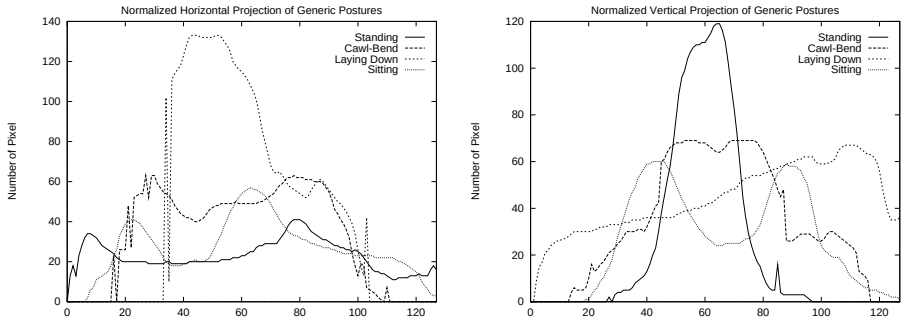


Figure 20: The vertical and horizontal normalized projections of standing, crawling-bending and laying down postures used in the body posture estimation

5.2 Estimation of the human body posture

People could be in many different postures while they are performing actions. Each posture has different appearances (views) varying with the point of view. Our system makes the assumption that the angle between the view direction and the ground plane is within -45 to +45 degrees. We collected examples of people over a wide range of views, and extracted their silhouettes to discover the order of body parts on the silhouette for different postures. We observed that four different main postures (standing, sitting, crawling-bending and laying down) have large differences in the order of body parts. The order in other postures is typically a variation of one of the main postures. *Ghost* classifies the observed human body posture in a *hierarchical* manner. Any body posture is classified into one of the four *main postures* (standing, sitting, crawling-bending, laying) and then each main posture is classified into one of three *view-based appearances* (front-view, left-side view and right-side view).

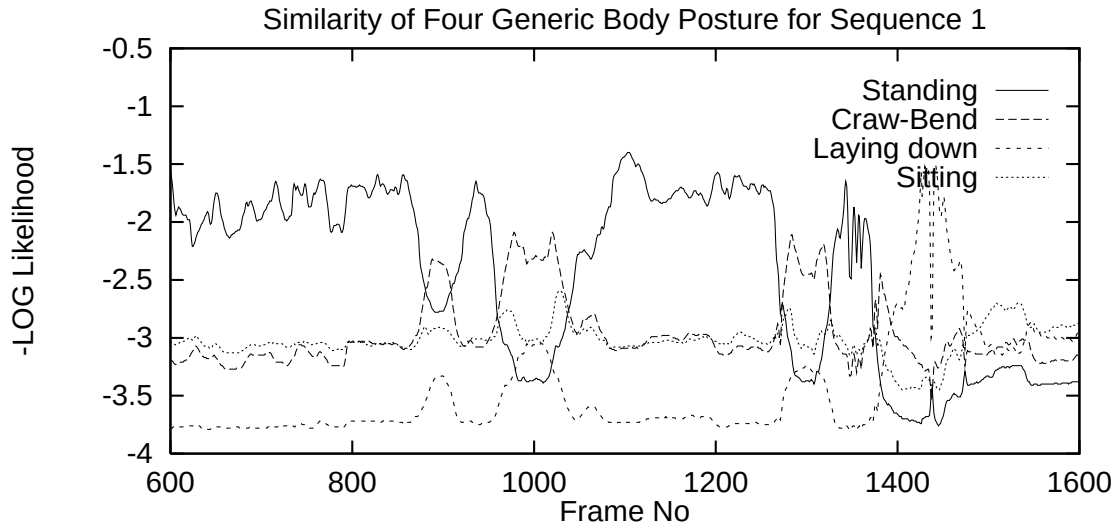
A body posture is represented by the normalized horizontal and vertical projections (projection histograms) of its silhouette, the median coordinate, and the major axis. Average normalized horizontal and vertical projection templates for each main posture (and for each view-based appearance of each main posture) were computed experimentally using 4500 silhouettes of 7 different people in 3 different views. These features are used to determine the similarity of the given posture to one of the four main postures. In Figure 20, the normalized vertical and horizontal projection templates of standing, crawling-bending, laying down, and sitting postures used in the body posture estimation in *Ghost* are shown.

Ghost determines and normalize the vertical and horizontal projections of the silhouette. Normalization is done by rescaling the silhouette into a fixed vertical length while keeping the original aspect ratio. *Ghost* then compares the observed silhouette with the projection templates of the four main postures using the sum of absolute difference method to estimate the most similar main posture. Let S^i be the similarity between the detected silhouette and ith main posture, H^i and V^i the horizontal and vertical projections of ith main posture,respectively, and P and R the horizontal and vertical projections of the detected silhouette. S_i is calculated as:

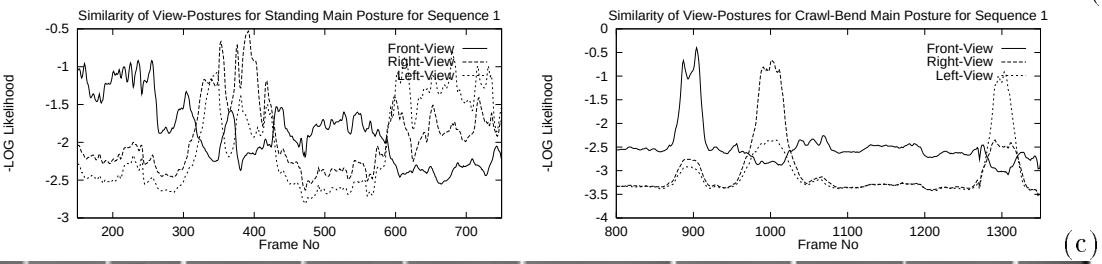
$$S_i = -log(\sum_{h}^{128}\sum_{v}^{128} |(H_h^i - P_h)| + |(V_v^i - R_v)| \quad (3)$$

Ghost determines the most similar main posture by using the highest score; it then applies the same method to determine the most similar view-based appearance for the estimated main posture. In Figure 21, the result of the main posture and the view-based appearance estimation are shown for two

(a)

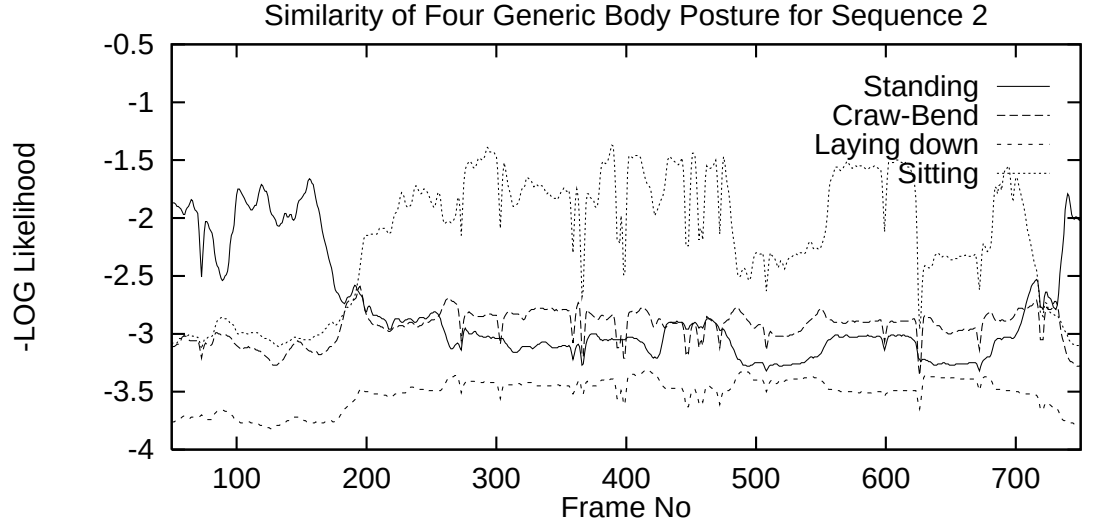


(b)



(c)

(d)



(e)

Figure 21: The similarity of four main postures for two different sequences

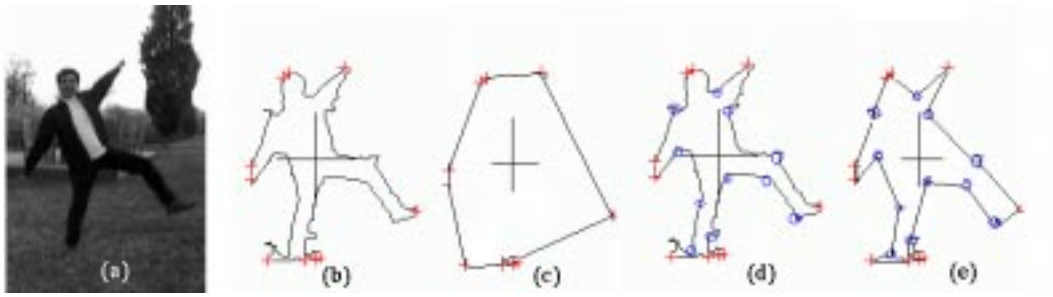


Figure 22: The body in the standing pose (a), the convex-hull vertices (b), the shape approximation by the convex-hull vertices (c), the convex and concave hull vertices (d), and the shape approximation by convex and concave hull vertices (e).

sequences. In sequence 1 (1750 frames), the person performs some simple work-out actions. He was in the following postures (with frame numbers): standing (0-850), crawling-bending (850-910), standing (920-970), crawling-bending(left view) (970-1080), standing (1080-1270), crawling-bending (right view) (1270-1320), standing (1320-1380) and laying down (1400-1470). The graph in Figure 21(b) shows how the estimation method is able to select the correct posture over time. 95% of the postures were successfully estimated in that sequence. Figure 21(c) shows the view-based appearance estimation for the standing (left) and crawling-bending (right) main postures for sequence 1. Note that when the body is in the crawling-bending posture (the peaks in the figure 21(c)-right), the view-based appearance was successfully estimated. In sequence 2 (750 frame), the person performs a "walk-sit-talk" action. She was in the following postures: standing (0-180), sitting (200-700) and standing (710-750). The graph in Figure 21(e) shows estimation results for sequence 2. 98% of the postures are correctly estimated in that sequence.

5.3 Detection of the convex and concave hulls on silhouettes

Because of the topology of humans, it is likely that some body parts always appear on the extreme points or sharp curvatures of the silhouette boundary. We need to find those points on the silhouette in order to find the set of locations which will be labeled as body parts. We implemented a recursive convex hull algorithm (Graham scan) to find these vertices (hulls) which are candidates to belong to one of the primary body parts. We modified the Graham scan convex-hull algorithm to be able to exploit the silhouette property and speed up the calculations. Convex hull vertices alone are not enough for shape representation of the silhouette. We need also some concave hull vertices to get a better representation of the silhouette. Therefore, we implemented it in a recursive manner, such that we first find the convex hull vertices of the silhouette in the first iteration, and then we applied the same convex hull algorithm to each of the silhouette segments between two convex hull vertices detected in first iteration to find the concave hull vertices. Figure 22 shows one of the standing poses (a), the convex hull vertices on its silhouette(b), the shape approximation by the convex hull vertices alone (c), the convex and concave hull vertices on its silhouette(d), and the shape approximation by convex and concave hull vertices(e).

5.4 Prediction of body part locations

5.4.1 Prediction of the head location

We need to select one body part as a starting point to be able use the order constraints and relative distances of body parts, If we know the location of at least one body part, we can predict the other

Figure 23: the silhouette segments for the head location predicted by *Ghost*

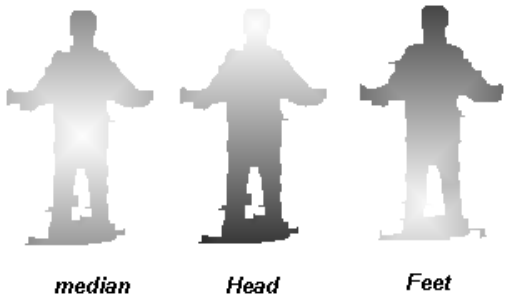


Figure 24: Distance transform from the head, feet, and median

parts with respect to that part. We take the head as a reference point to locate the other parts. The head is a stable body part compared to the others, and its location can be easily predicted. *Ghost* tries to find a silhouette segment which includes the head by combining the constraints on the order of body parts on the silhouette for the expected main posture, the principal axis of the silhouette and the ground plane information. Let p be major axis of the silhouette, and let l_1 and l_2 be the two lines which are intersecting with p at the median coordinate of the silhouette. The angles between l_1 and p is 45 degree and the angle between l_2 and p is -45 degree. *Ghost* determines the silhouette segments whose starting and end points intersect points between the silhouette and lines l_1 and l_2, respectively. The ground plane is used to eliminate the silhouette segment which is on the opposite side of the head with respect to the median coordinate. In Figure 23, the silhouette segments for the predicted head locations are shown.

5.4.2 Prediction of the feet and hands and torso locations

After *Ghost* determines the head location, it tries to find the other primary body parts in the order of the feet, the hands, and the torso using prior-knowledge about the topology of the estimated body posture. Let M^i be a subset of primary and secondary body part which are predicted to be visible given restimated main posture. Let C^t be set of the convex and concave hull vertices detected on the silhouette boundary. We need to find a fast and simple partial mapping from M^i to C^t to solve the labeling problem; the labeling should be consistent with the order of body parts for the estimated posture. Relative path distances are used to check if the mapping is consistent with the order. The relative distances and the order-constraints for each main posture and view-based appearance of the main posture are calculated experimentally. Initially, the path distances from the head and median to hull vertices are computed using a distance transform method. After the initial distance calculation is

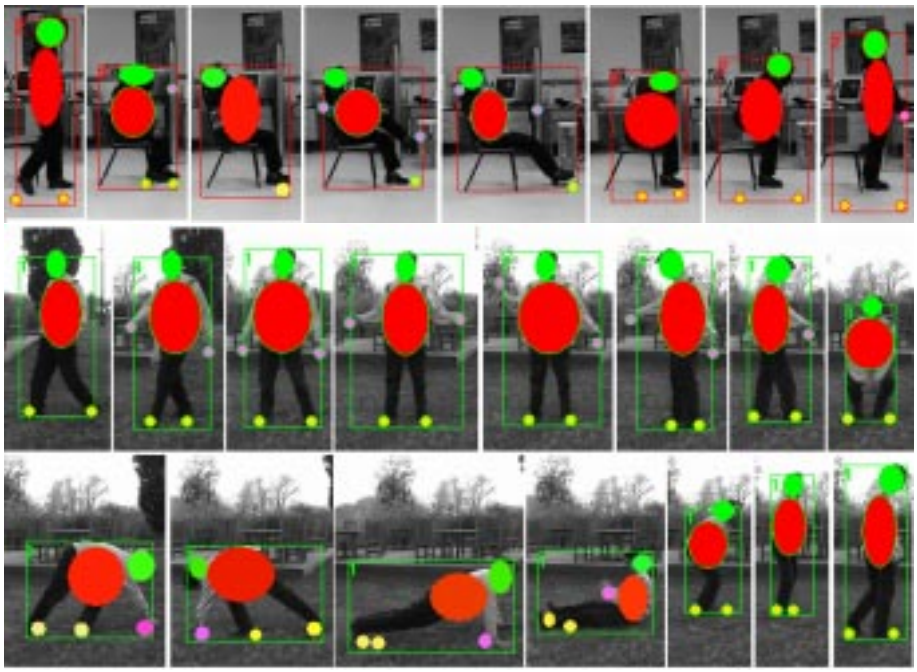


Figure 25: Examples of using silhouette model to locate the body parts in different actions

done, each hull vertex is assigned to one or more body parts according to their relative distances as long as they satisfy the order-constraints of the estimated posture. The hull vertices which are assigned to the feet are labeled first, then the hull vertices for shoulders, upper backs, hands, elbows, armpits and knees are labeled. If there is more than one vertex assigned to the same body part, another search is applied by narrowing the order constraints for those vertices. While labeling a vertex, the distances to the previously labeled vertices should also be consistent. Torso is located to region between the median and the head along the major axis of the silhouette. Examples of the body part labeling results of *Ghost* are shown in Figure 25

6 Future Work

In this chapter, we will discuss the work we are planing to do to extend W^4.

6.1 Background Modeling

In our current implementation, the background modeling has some limitations. It requires that no moving object be in the scene during determination of the background statistics. We should relax that assumption.

Currently, the background model is updated periodically by calculating the background statistics for the last 60 frames and updating the background model where there is no motion during that period. The previously used background statistics are still used where there is motion during last 60 frames. This type of background updating need some improvements, because of the following observations, It cannot recover the background statistics for a region that is initially a background object, but then moves (e.g., a car) and create a new background region. When a new object enters the scene, it will be detected and tracked. If this object does not move for a long time, it should be included in the background model by updating the background statistics of the region that object occupies.

6.2 Body Pose Estimation and Labeling of The Body Parts

There are several directions that we are pursuing to improve the performance of *Ghost* and to extend its capabilities. We are studying to use of a statistical model to formulate the partial mapping from the hulls on the boundary to the body part. The labeling problem can be possibly solved by using a second order Markov Model. The relative motion of the body parts could be employed to get more robust and correct mappings. In our current implementation, shadows create problems when locating the body parts which are close to the ground. Better results could be obtained if shadows are removed from the silhouette.

6.3 Human Identification

In our current implementation of W^4, foreground regions detected in the scene are assumed to belong to a person, then further body part analysis is applied them. However, this assumption sometimes fails when the scene contains other objects, such as cars, briefcases, boxes or animals. Therefore, W^4 needs an identification system to distinguish people from other objects.

6.4 Activity Recognition

W^4 will be extended with models to recognize the actions of the people it tracks. Specifically, we are interested in interactions between people and objects- e.g., people exchanging objects, leaving objects in the scene, taking objects from the scene. The description of the people-their global motion and the motion of the their body parts- developed by W^4 are designed to support such activity recognition.

6.5 Restricted Camera Motion and "Stop and Look" approach

W^4 currently operates on video taken from a stationary camera, and many of its image analysis algorithms would not generalize easily to images taken from a moving camera. We are planning to add restricted camera motion for extending the monitoring area and tracking the target person over a wide

range. We will extend W^4 to be able to track a person in a wide-range by panning the camera and updating the background model.

References

[1] K. akita. mage sequence analysis of real world human motion. *Pattern Recognition*, 17(4):73–83, 1984.

[2] A. Azarbayejani and A. Pentland. Camera self-calibration from one-point correspondence. Technical Report 341, MIT Media Lab Perceptual Computer Section, 1996.

[3] A. Azarbayjani, C. Wren, and A. Pentland. Real-time 3-d tracking of the human body. In *Proc. IMAGE'COM 96*, Bordeaux, France, 1996.

[4] S. Barnard and M. Fischer. Computational stereo. *Computing Survey*, 14(4):553–572, 1982.

[5] A. Bobick and J. Davis. Real-time recognition of activity using temporal templates. In *IEEE Workshop on Application of Computer Vision*, pages 1233–1251, 1996.

[6] A. Bobick, J. D. S., Intille, F. Baird, L. Cambell, Y. Irinov, C. Pinhanez, and A. Wilson. Kidsroom: Action recognition in an interactive story environment. Technical Report 398, M.I.T Perceptual Computing, 1996.

[7] R. Bolles and J.Woodfill. Spatiotemporal consistency checking of passive range data. In *International Symposium on Robotics Research*, 1993.

[8] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complexe action recognition. Technical report, 1997.

[9] C. Bregler. Learning and recognizing human dynamics in video sequences. In *Proceedings of the Computer Vision and Pattern Recognition*, pages 569–574, 1997.

[10] Q. Chai and J. K. Aggarwal. Tracking human motion using multiple cameras. In *International Conference on Patern Recognition*, pages 68–72, 1996.

[11] J. Costeria and T. Kanade. A multi-body factorization method for motion analysis. Technical Report CMU-CS-TR-94-220, Scholl of Computer Science,Carnegie Mellon Univeristy, 1994.

[12] J. D. Courtney. Automatic video indexing via object motion analysis. *Pattern Recognition*, 1997.

[13] T. Darrel and A. Pentland. Recognition of space-time gestures using distibuted representation. Technical Report 197, M.I.T Perceptual Compuring, 1993.

[14] O. Faugeras. *Three-Dimensional Computer Vision-Chapters: 2,3,6.* 1993.

[15] S. Fejes and L. Davis. Eploring visual motion using projections of flow fields. In *the DARPA Image Understanding Workshop*, pages 113–122, 1997.

[16] P. Fua. A parallel stereo algorithm that produces dense depth maps and preverse image features. In *Machine Vision and Applications*, 1993.

[17] I. Haritaoglu, D. Harwood, and L. Davis. W4: Who, when, where, what: A real time system for detecting and tracking people. In *Face and gesture Recognition Conference*, 1998.

[18] S. Haykin. *Adaptive Filter Theory*. Prentice-Hall, 1991.

[19] G. Herzog and K. Rohr. Integrating vision and langauge: Towards automatic description of human movements. Technical Report 122, Univerisitat des Saarlandes, 1995.

[20] S. Intille and A. Bobick. Visual tracking using closed-worlds. Technical Report 294, MIT Media Lab Perceptual Computer Section, 1994.

[21] S. Intille, J. Davis, and A. Bobick. Real-time closed-world tracking. In *Proceedings of the Computer Vision and Pattern Recognition*, pages 697–703, 1997.

[22] S. Iwasawa, K. Ebihara, J. Ohya, and S. Morishima. Real-time estimation of human body posture from monocular thermal images. In *Proceedings of the Computer Vision and Pattern Recognition*, 1997.

[23] T. Jebara and A. Pentland. Parameterized structure from motion for 3d adaptive feedback tracking of faces. Technical Report 401, MIT Media Lab Perceptual Computer Section, 1997.

[24] T. Kanade. Stereo machine for video rate dense depth mapping and its new application. In *Proceedings of the Computer Vision and Pattern Recognition*, 1996.

[25] T. Kanade, H. Kano, and S. Kimura. Development of a video-rate stereo machine. Technical report, Robotics Institute, Carnegie Mellon Univeristy, 1997.

[26] K. Konolige. Small vision systems: hardware and implementation. In *International Symposium of Robotics Research*, 1997.

[27] Y. Kuno, T. Watanabe, Y. Shimosakoda, and S. Nakagawa. Automated Detection of Human fro Visual Surveillance System. In*Proceedings in International Conference of Pattern Recognition*, 1996.

[28] M. Leung and Y. H. Yang. Human body motion segmentation in a complex scene. *Pattern Recognition*, 20(1):55–64, 1987.

[29] M. Leung and Y. H. Yang. A region based approach for human body motion analysis. *Pattern Recognition*, 20(3):321–339, 1987.

[30] M. Leung and Y. H. Yang. A model based approach to labeling human body outlines. 1994.

[31] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proc. Royal Society*, 204(1):301–328, 1979.

[32] T. Olson and F. Brill. Moving object detection and event recognition algorithms for smart cameras. *Proc. DARPA Image Understanding Workshop*, pages 159–175, 1997.

[33] M. Oren, C. papageorgiou, E. Osuna, and T. Poggio. Pedestraian detection using wavelet templates. In *Proc. Computer Vision and Pattern recognition*, Puerto Rico, 1997.

[34] J. O'Rourke and N. Badler. Model-based image analysis of human motion using constraint propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2:522–536, 1980.

[35] A. Pentland and A. Liu. Modelling and prediction of human behavior. Technical Report 433, M.I.T Media Lab Perceptual Computing, 1995.

[36] A. Pentland and A. Liu. Modeling and prediction of human behavior. Technical Report 403, MIT Media Lab Perceptual Computer Section, 1997.

[37] R. Polana. *Temporal Texture and Activity Recognition*. PhD thesis, Univ. of Rochester, 1994.

[38] R. Polana and R. Nelson. Recognizing activities. In *International Conference on Patern Recognition*, 1994.

[39] R. Polana and R. Nelson. Nonparametric recognition of nonrigid motion. Technical Report 575, Univ. of Rochester, 1995.

[40] R. Polona and R. Nelson. Low level recognition of human motion. In *Non Rigid Motion Workshop*, pages 21–29, 1994.

[41] J. Segen and S. Pingali. A camera-based system for tracking people in real-time. In *ICCV*, 1996.

[42] S. M. Seitz and C. R. Dyer. View-invariant analysis of cyclic motion. *International Journal of Computer Vision*, 25:1–25, 1997.

[43] Y. Y. Shanon Xuan Ju, Michael J. Black. Cardboard people: A parameterized model of articulated image motion. In *Second International Conference on Automatic Face and Gesture Recognition*, Killington, Vermont, 1996.

[44] T. E. Starner. Visual recognition of american sign language using hidden markov models. Master's thesis, MIT Media Lab Perceptual Computer Section, 1995.

[45] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[46] T. Westman, D. Harwood, T. Laitinen, and M. Pietikainen. Color segmentation by hierarchical connected components analysis with image enhancement by symmetric neighborhood filters. In *Proceedings of the Computer Vision and Pattern Recognition*, 1990.

[47] A. Wilson, A. Bobick, and J. Cassell. Recovering the temperol structure of natural gesture. In *Face and gesture Recognition*, 1996.

[48] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. Technical Report 353, MIT Media Lab Perceptual Computer Section, 1995.

[49] C. Wren and A. Pentland. Dynamic modeling of human motion. Technical Report 415, MIT Media Lab Perceptual Computer Section, 1997.

[50] Y. Yacoob and M. Black. Parameterized modelling and recognition of activities. In *Proc. International Conference of Computer Vision*, Mumbai-India, 1998.

[51] J. Yamato, J. Ohya, and K. Ishii. Recognizing human actions in time-sequential images using hidden markov model. In *Proceedings of the Computer Vision and Pattern Recognition*, pages 379–385, 1992.

[52] Zabih and J. Woodfill. Spatiotemporal consistency checking of passive range data. In *European Conference on Computer Vision*, 1994.

7 Reading List

7.1 Motion and Stereo

1. P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1989.

2. S. Barnard and M. Fischer. Computational stereo. *Computing Survey*, 14(4):553–572, 1982.

3. J. Barron, D. Fleet, S. Beauchemin, and T. Burkitt. Performance of optical flow techniques. In *Proceedings of the Computer Vision and Pattern Recognition*, pages 236–242, 1992.

4. M. Black and P. Anandan. A framework for the robust estimation of optical flow. In *ECCV*, 1992.

5. J. Costeria and T. Kanade. A multi-body factorization method for motion analysis. Technical report, Scholl of Computer Science,Carnegie Mellon Univeristy, 1994

6. O. Faugeras. *Three-Dimensional Computer Vision-Chapters: 2,3,6*

7. E. Grimson. Computational experiments with a featue based stereo algorithm. *Pattern Analysis and Machine Inteligence* 7(1), 1985

8. B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

9. A. Jepson and M. Black. Mixture models for optical flow computation. Technical Report RBCV-TR-93-44, Univerity of Toronto, 1993.

10. T. Kanade, at al. Development of a video-rate stereo machine. *Proc. of International Robotics and Systems Conference*, Pittsburg, 1995

11. K. Nishihara. Practical real-time imaging stereo matcher. *Optical Engineering*, 23(5):536–545, 1984.

12. C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

7.2 Human Tracking

1. A. Azarbayejani and A. Pentland. Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. Technical Report 363, MIT Media Lab Perceptual Computer Section, 1996.

2. P. Fieguth and D. Terzopoulos. Color-based tracking of heads and other mobile objects at video frame rates. In *CVPR*, pages 21–27, 1997.

3. D. P. Huttenlocker and J. J. Noh. Tracking Non-rigid objects in complex scenes. Technical Report TR92-1320, Cornell University, 1992

4. S. Intille, J. Davis, and A. Bobick. Real-time closed-world tracking. Technical Report 403, MIT Media Lab Perceptual Computer Section, 1996.

5. Y. Kuno, T. Watanabe, Y. Shimosakoda, and S. Nakagawa. Automated detection of human for visual surveillance system. In *International Conference on Patern Recognition*, 1996.

6. F. Liu and R. Picard. Detecting and segmenting period motion. Technical Report 400, MIT Media Lab Perceptual Computer Section, 1997

7. M. Oren C. Papageorgiou, P Sinha, T. Poggio. Pedestraian detection using wavelet templates. In *Proc. Computer Vision and Pattern recognition*, 1997.

8. N. Oliver, A. Pentland, and F. Berand. Lafter: Lips and face real time tracker. Technical Report 390, MIT Media Lab Perceptual Computer Section, 1996.

9. T. Olson and F. Brill. Moving object detection and event recognition algorithms for smart cameras. *Proc. DARPA Image Understanding Workshop*, pages 159–175, 1997.

10. R. Polana and R. Nelson. Nonparametric recognition of nonrigid motion. Technical Report 575, Univ. of Rochester, 1995.

11. J. Rehg and T. Kahande. Digiteyes: Vision-based human hand tracking. Technical Report CMU-CS-93-220, 1993.

12. S. Smith. Asset-2: Real-time motion segmentation and object tracking. Technical report, Oxford University, 1995.

13. C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. Technical Report 353, MIT Media Lab Perceptual Computer Section, 1995.

7.3 Activity Recognition

1. A. Bobick and J. Davis. An appearance-based representation of action. Technical Report 369, M.I.T, 1996.

2. M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complexe action recognition. Technical report, 1997.

3. C. Bregler. Learning and recognizing human dynamics in video sequences. In *Proceedings of the Computer Vision and Pattern Recognition*, 1997.

4. C. Cedras and M. Shah. Motion-based recognition: A survey. *Image and Vision Computing*, 13(2):129–155, 1995.

5. T. Darell and A. Pentland. Recognition of space-time gestures using a distributed representation. Technical Report 197, M.I.T Media Lab Perceptual Computer Section, 1993

6. J. W. Davis and A. F. Bobick. The representation and recognition of action using temporal templates. Technical Report 402, MIT Media Lab Perceptual Computer Section, 1996.

7. A. Liu and A. Pentland. Modelling and prediction of human behavior. Technical Report 433, M.I.T Media Lab Perceptual Computing, 1995.

8. A. Pentland. Machine understanding of human action. Technical Report 350, MIT Media Lab Perceptual Computer Section, 1995.

9. R. Polona and R. Nelson. Recognition of activities. *International conference of pattern recognition, p:815-820 1994*

10. T. Starner. Visual recognition of american sign language using hidden markov models. Master's thesis, 1995.

11. A. Wilson, A. Bobick, and J. Cassell. Recovering the temperol structure of natural gesture. In *Proc. of Face and gesture Recognition*, 1996.

12. Y. Yacoob and M. Black. Parameterized modelling and recognition of activities. In *Proc. International Conference of Computer Vision*, Mumbai-India, 1998.

13. J. Yamato, J. Ohya and K. Ishii. Recognizing Human Actions in Time-Sequential Images Using Hidden Markov Model. Proc. CVPR, 379-385,1992