W^4S: A Real-Time System for Detecting and Tracking People in $2\frac{1}{2}D$

Ismail Haritaoglu, David Harwood and Larry S. Davis
Computer Vision Laboratory
University of Maryland
College Park, MD 20742, USA

Abstract. W^4S is a real time visual surveillance system for detecting and tracking people and monitoring their activities in an outdoor environment by integrating realtime stereo computation into an intensity-based detection and tracking system. Unlike many systems for tracking people, W^4S makes no use of color cues. Instead, W^4S employs a combination of stereo, shape analysis and tracking to locate people and their parts (head, hands, feet, torso) and create models of people's appearance so that they can be tracked through interactions such as occlusions. W^4S is capable of simultaneously tracking multiple people even with occlusion. It runs at 5-20 Hz for 320x120 resolution images on a dual-pentium 200 PC.

1 Introduction

W^4S is a real time system for tracking people and their body parts in monochromatic stereo 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 in $2\frac{1}{2}D$ so that it can track people (*Who?*) through occlusion events in the imagery. W^4S represents the integration of a real-time stereo (SVM) system with a real-time person detection and tracking system (W^4[14]) to increase its reliability. SVM [12] is a compact, inexpensive realtime device for computing dense stereo range images which was recently developed by SRI. W^4 [14] is a real time visual surveillance system for detecting and tracking people in an outdoor environment using only monochromatic video.

In this paper we describe the computational models employed by W^4S to detect and track people. These models are designed to allow W^4S 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^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. W^4S builds "appearance" models of people so that they can be identified after occlusions or after interactions during which W^4S cannot track them individually. 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.

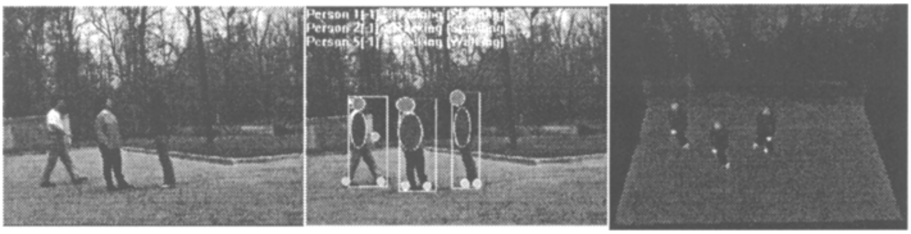

Fig. 1. Examples of detection result: intensity image (left), detected people and their body parts form the intensity only (middle) and their placement in the $2\frac{1}{2}D$ scene by W^4S (right)

W^4S has been designed to work with only visible monochromatic video sources. While most previous work on detection and tracking of people has relied heavily on color cues, W^4S is designed for outdoor surveillance tasks, and particularly for low light level situations. In such cases, color will not be available, and people need to be detected and tracked based on weaker appearance, motion, and disparity cues. W^4S is a real time system. It currently is implemented on a dual processor Pentium PC and can process between 5-20 frames per second depending on the image resolution and the number of people in its field of view.

In the long run, W^4S 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 descriptions of people - their global motions and the motions of their parts - developed by W^4S are designed to support such activity recognition.

W^4S 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 [10]. At this point, the surveillance

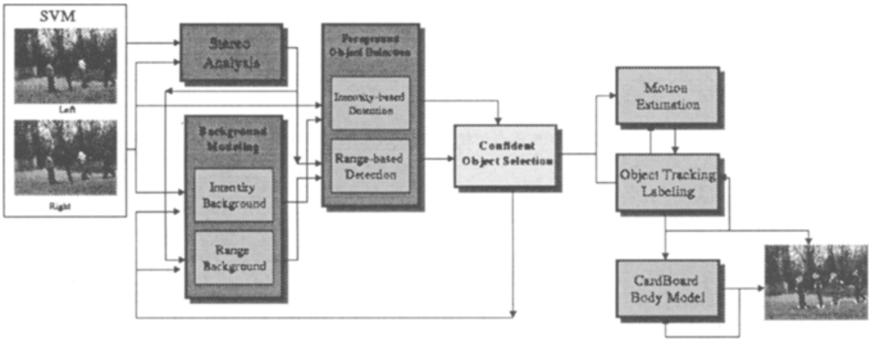

Fig. 2. Detection and Tracking System

system might stop and invoke a system like W^4S to verify the presence of people and recognize their actions.

The system diagram of W^4S is shown in Figure 2. 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. These algorithms are described in Section 4. 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. These algorithms are described in Section 6. 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.

2 Related Work

Pfinder [1] is a real-time system for tracking a person which uses a multi-class statistical model of color and shape to segment a person from a background scene. It finds and tracks people's head and hands under a wide range of viewing condition.

[6] is a general purpose system for moving object detection and event recognition where moving objects are detected using change detection and tracked using first-order prediction and nearest neighbor matching. Events are recognized by applying predicates to a graph formed by linking corresponding objects in successive frames.

KidRooms [3, 9] 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.

Bregler uses many levels of representation based on mixture models, EM, and recursive Kalman and Markov estimation to learn and recognize human dynamics [5]. Deformable trackers that track small images of people are described in [7].

Realtime stereo systems have recently become available and applied to detection of people. Spfinder[2] 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 [13] has implemented a Z-keying method, where a subject is placed in correct alignment with a virtual world. SRI has been developing a person detection system which segments the range image by first learning a background range image and then using statistical image compression methods to distinguish new objects [12], hypothesized to be people.

3 Stereo Analysis

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 [12]. 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

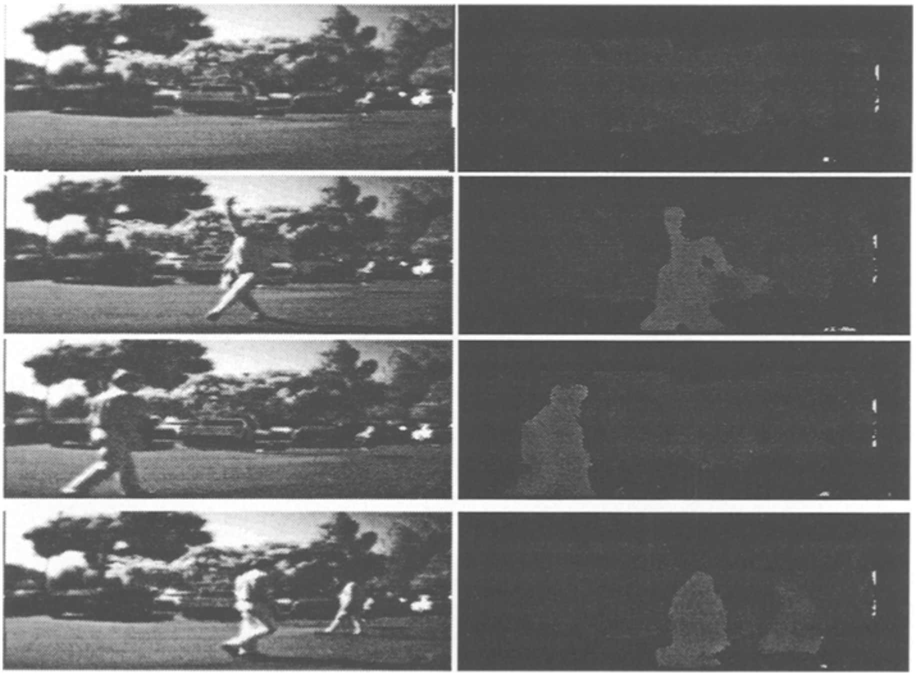

Fig. 3. Examples of the range images calculated by area correlation stereo

utilized by W^4S, with the stereo algorithm running on the host PC, to obtain better quality disparity images.

Figure 3 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.

4 Background Scene Modeling and Foreground Region Detection

The background scene is modeled and the foreground regions are detected in both the intensity image and the disparity image simultaneously. Frame differencing in W^4S is based on a model of background variation obtained while the scene contains no people. The background scenes for the intensity images (the disparity images) are modeled by representing each pixel by three values; its minimum and maximum intensity (disparity) values and the maximum intensity (disparity) 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^4S determines to contain no 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. 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 following detection method is explained only for the intensity images; it is the same for the disparity images.

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^4S 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^4S applies morphological operators to foreground pixels only after noise pixels are eliminated. So, W^4S 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.

5 Foreground Region Selection

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.

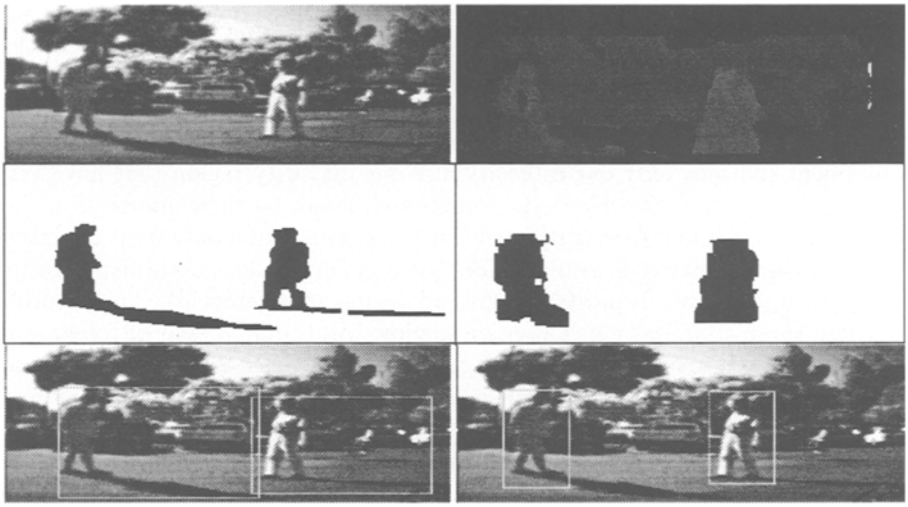

Fig. 4. An example showing that how stereo-based detection eliminates shadows during object detection.

- 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

- 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 4 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 (≥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.

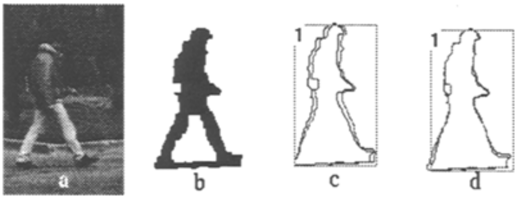

Fig. 5. Motion estimation of body using Silhouette Edge Matching between two successive frame a: input image; b: detected foreground regions; c: alignment of silhouette edges based on difference in median; d: final alignment after silhouette correlation

6 Object Tracking

The goals of the object tracking stage are to:

- 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^4S.
- employ tracking algorithms to estimate the position (of the torso) of each object, and update the motion model used for tracking. W^4S 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^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 6 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.

W^4S 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^4S tracks objects using different methods under each condition.

6.1 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 monitored and tracked.

6.2 Tracking

Here, we consider the situation that an object continues to be tracked as a single foreground region. W^4S 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^4S 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^4S uses a two stage matching strategy to update its global position estimate of an object. The

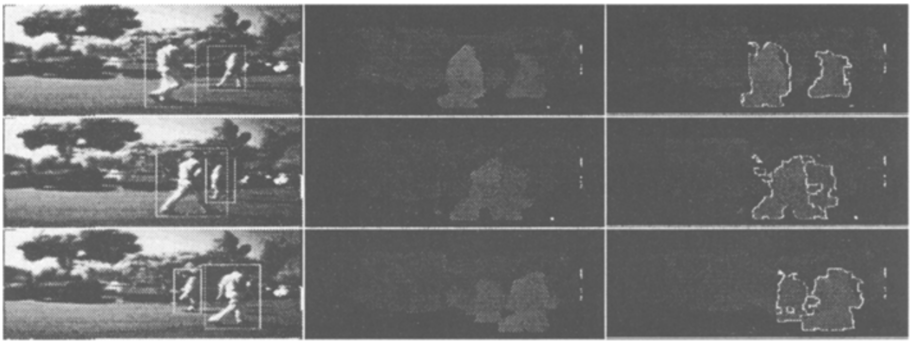

Fig. 6. An Example how range data is useful to track the people by segmenting the range data during occlusion

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 5.

6.3 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. In this case, one object will be matched to two or more current foreground regions. W^4S 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^4S can track the constituent objects over several frames, then it assumes that they are separate objects and begins to track them individually.

6.4 Region merging

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

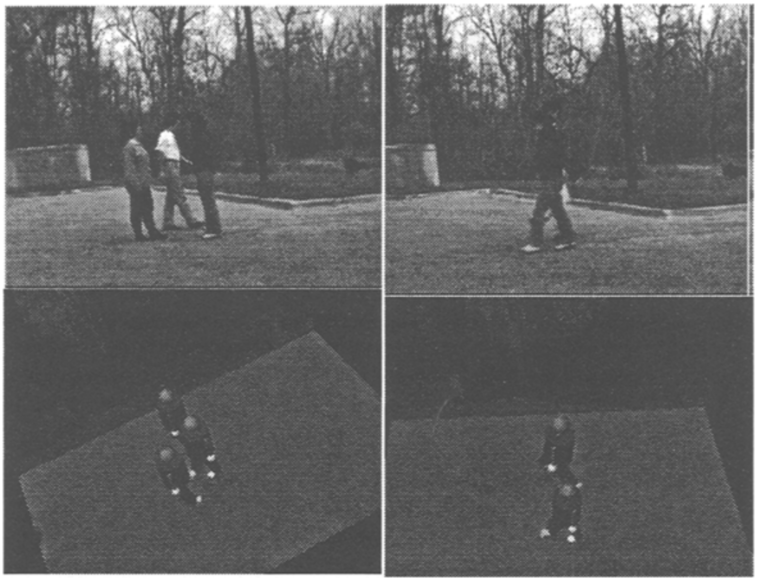

Fig. 7. W^4S uses range data to detect the people during occlusion and determine their relative location in $2\frac{1}{2}D$

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 [15] is used for segmentation. The disparity segmentation can successfully distinguish the objects when they are at different ranges as shown in figure 6. 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 7 shows two examples of W^4S tracking people by illustrating their locations in $2\frac{1}{2}D$.

A problem that arises when the merged region splits, and the people "reappear", 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^4S uses two types of appearance models that it constructs while it is tracking an isolated person.

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

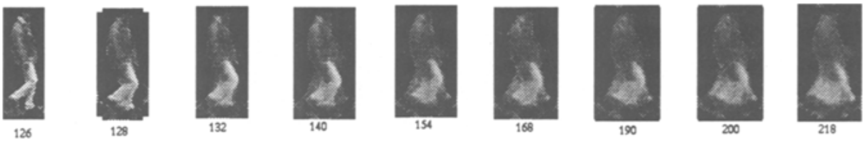

Fig. 8. An example of how temporal templates are updated over time

object is defined by:

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

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 8.

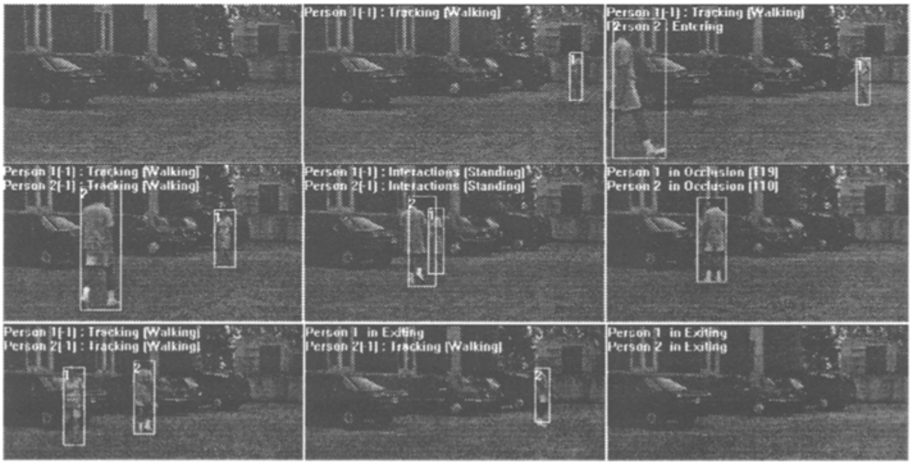
Person 1[-1] : Tracking [Walking]
Person 1[-1] : Tracking [Walking]
Person 2 : Entering
Person 1[-1] : Tracking [Walking]
Person 2[-1] : Tracking [Walking]
Person 1[-1] : Interactions [Standing]
Person 2[-1] : Interactions [Standing]
Person 1 in Occlusion [119]
Person 2 in Occlusion [110]
Person 1[-1] : Tracking [Walking]
Person 2[-1] : Tracking [Walking]
Person 1 in Exiting
Person 2[-1] : Tracking [Walking]
Person 1 in Exiting
Person 2 in Exiting

Fig. 9. An example for W^4S tracking; two people are entering, walking, meeting and leaving

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

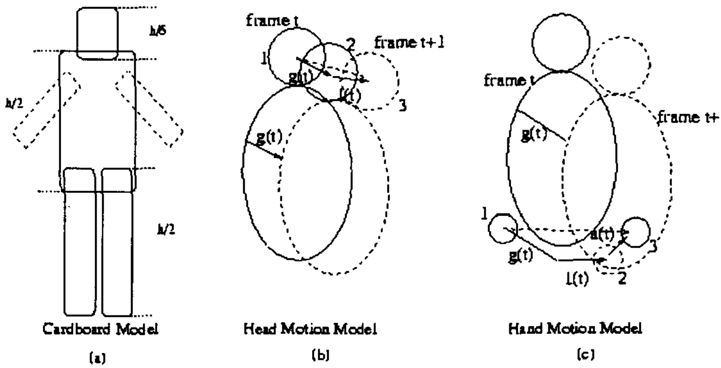

Fig. 10. Cardboard model used in W^4S (a), and motion models used for the head (b) and hands (c).

7 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^4S 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* [11] 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 Model can be used to predict the positions of the individual body parts from frame to frame. Figure 10 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 10(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

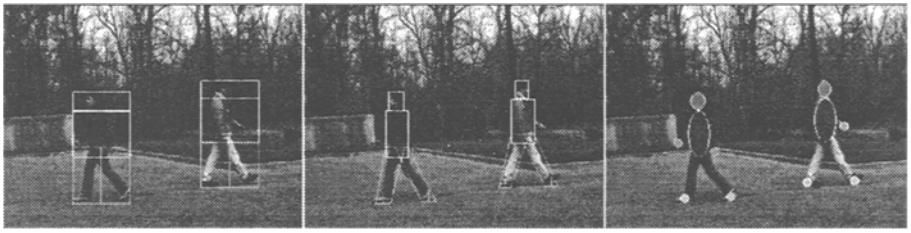

Fig. 11. An example of Cardboard Model to show How Head, Torso, Legs Hands and Feet are Located. Initial Bounding boxes are located on foreground regions (a); Cardboard model analysis locates the body part (b); illustration of body part location by ellipsis (c)

is taken as the 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 11. 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 11 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.

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.

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 Figure 13 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

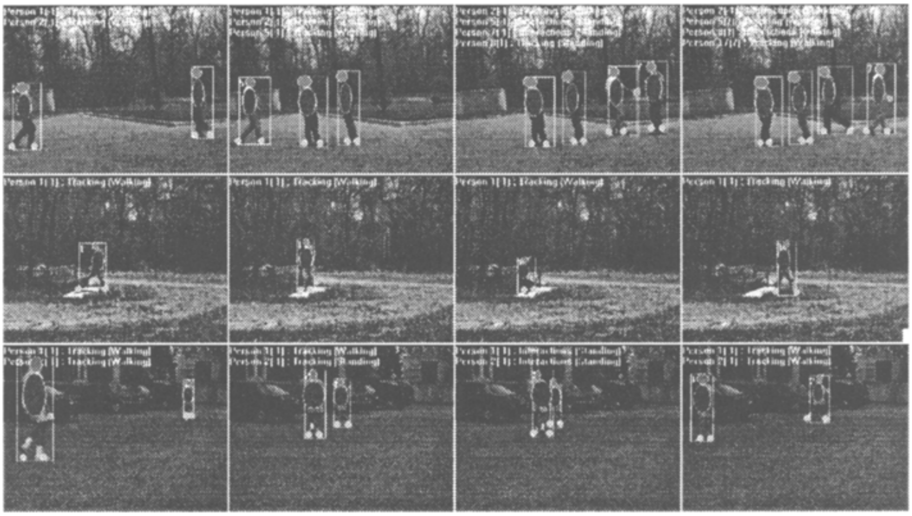

Fig. 12. Examples of using the cardboard model to locate the body parts in different actions: four people meet and talk (first line), a person sits on a bench (second line), two people meet (third line).

which the left hand is occluded by the body, and they have significantly worse correlation scores (higher SAD).

8 Discussion

We described a real time visual surveillance system (W^4S) for detecting and tracking people and monitoring their activities in an outdoor environment by integrating realtime stereo computation into an intensity-based detection and tracking system. W^4S has been implemented in C++ and runs under the Windows NT operating system. Currently, for 320x120 resolution images, W^4S runs at 20 Hz on a PC which has dual 200 Mhz pentium processor. It has the capa-

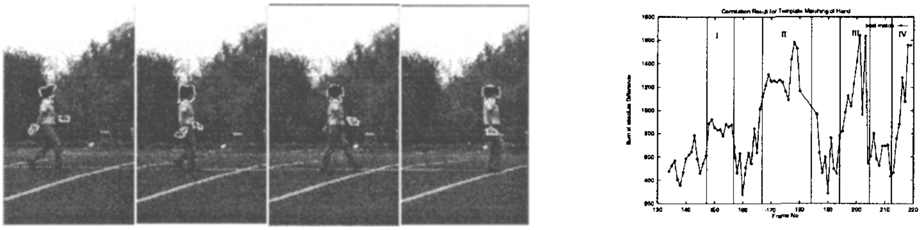

Fig. 13. An Example of hands and head tracking of a walking person and correlation results for the left hand of that person.

bility to track multiple people against complex background. Figure 12 illustrates some results of the W^4S system in scenes of parking lots and parkland.

There are several directions that we are pursuing to improve the performance of W^4S and to extend its capabilities. Firstly, the cardboard model used to predict body pose and position is restricted to upright people. We would like to be able to recognize and track people in other generic poses, such as crawling, climbing, etc. We believe this might be accomplished based on an analysis of convex hull-like representations of the silhouettes of people. Finally, our long term goal is to be able to recognize interactions between the people that W^4S is tracking. We are studying the use of temporal logic programs for the representation of actions, and to control the application of visual routines to peoples movements.

References

1. C. Wren, A. Azarbayejani, T. Darrell, A. Pentland "Pfinder: Real-Time Tracking of the Human Body", *In Proc. of the SPIE Conference on Integration Issues in Large Commercial Media Delivery Systems*, October 1995.
2. A. Azarbayejani, C. Wren and A. Pentland "Real-Time Tracking of The Human Body" *In Proc. IMAGE COM 96*, Bordeaux, France, May 1996
3. S. Intille, J. Davis, A. Bobick "Real-Time Closed-Word Tracking", *In Proc. of CVPR*, pp.697-703, June 1997
4. A. F. Bobick and J. Davis "Real-Time recognition of activity using TemporalTemplates" *In Proc. Third IEEE Workshop on Application of Computer Vision*,pp.1233-1251, December, 1996
5. C. Bregler "Learning and Recognizing Human Dynamics in Video Sequences" *In Proc. CVPR 97*, pp.569-574, June 1997
6. T. Olson, F. Brill "Moving Object Detection and Event Recognition algorithms for Smart Cameras" *In Proc. DARPA Image Understanding Workshop*, pp.159-176, May 1997.
7. R. Polona, R. Nelson "Low Level Recognition of Human Motion", *In Proc. Non Rigid Motion Workshop*, November 1994.
8. A. Pentland "Machine Understanding Human Actions" *In Proc. DARPA Image Understanding Workshop*,pp.757-764, 1996.
9. A. Bobick, J. Davis, S. Intille, F. Baird, L. Cambell, Y. Irinov, C. Pinhanez, A. Wilson "KidsRoom: Action Recognition In An Interactive Story environment" *M.I.T. TR No: 398*, December 1996
10. S. Fejes, L.S. Davis "Exploring Visual Motion Using Projections of Flow Fields" *In. Proc. of the DARPA Image Understanding Workshop*, pp.113-122 New Orleans, LA,1997
11. S. Ju, M. Black, Y. Yacoob, "Cardboard People: A Parameterized Model of Articulated Image Motion", *International Conference on Face and Gesture Analysis*, 1996
12. K. Konolige "Small Vision systems: Hardware and Implementation", *Eighth International Symposium on Robotics Research*, Hayama, Japan, November, 1997
13. T. Kanade, "A Stereo Machine For Video Rate Dense Depth Mapping and Its New Application", *In Proc. CVPR*, 1996.
14. —, "W4: Who, When, Where, What: A Real Time System for Detecting and Tracking People" *Submitted to Face and Gesture Recognition Workshop, 1998.*
15. T. Westman, D. Harwood, T. Laitinen, M. Pietikainen "Color Segmentation by Hierarchical Connected Components Analysis With Image Enhancement by Symmetric Neighborhood Filters" *In Proc. CVPR*, pp:796-802, June, 1990