2008－CVIM－165 (18)
2008／11／27

A Novel Evidence Accumulation Framework for Robust Multi-Camera Person Detection and Tracking

Akio Kosaka[1),2)], Hidekazu Iwaki[1)], Takashi Miyoshi[1)],
Gaurav Srivastava[2)], Johnny Park[2)], and Avinash Kak[2)]

[1)]Future Creation Laboratory, Olympus Corporation, 2-3-1 Nishi-Shinjuku, Shinjuku, Tokyo 163-0914
[2)]Robot Vision Laboratory, Purdue University, West Lafayette, IN 47907
E-mails: {kosaka, hiwaki, gsrivast, jpark, kak}@purdue.edu
URL: http://cobweb.ecn.purdue.edu/RVL/

Abstract - We propose a novel evidence accumulation framework that accurately estimates the positions of humans in a 3D environment using camera networks. The framework consists of a network of distributed agents having different functionalities. The modular structure of the network allows scalability to large surveillance areas and robust operation. The framework does not assume reliable measurements in single cameras (referred to as 'sensing agents' in our framework) or reliable communication between different agents. There is a position uncertainty associated with single camera measurements and it is reduced through an uncertainty reducing transform that performs evidence accumulation using multiple camera measurements. Our framework has the advantage that single camera measurements do not need to be temporally synchronized to perform evidence accumulation. The system has been tested for detecting single or multiple humans in the environment. We conducted experiments to evaluate the localization accuracy of the position estimates obtained from the system by comparing them with the ground truth. We also developed a system capable of tracking and interacting with humans in motion to support human activities.

keywords: camera networks, distributed processing, evidence accumulation, uncertainty reduction.

1 Introduction

The largest advantage of multi-camera surveillance networks over single camera systems is their ability to combine information from different cameras into scene-level representations which give an enhanced awareness of the monitored environment. This ability depends critically on how the information is combined from the different cameras. We obviously need an evidence accumulation framework that is well-principled with regard to combining the uncertainties in the information gleaned from each camera. We also want such a framework to scale up easily, as more and more cameras are added to the network. As the camera network grows large, it is extremely difficult to synchronize the image capture by multiple cameras. Therefore, the framework should also be able to combine information from different cameras taking into account the uncertainty in image acquisition times. The goal of this paper is to present such an evidence accumulation framework suitable for practical scalable systems.

Our proposed framework consists of a hierarchy of agents. The lowest level of this hierarchy has 'sensing agents'; they perform extraction of candidate shapes and features. Higher levels of the agent hierarchy deal with 1) local accumulation of supporting evidence for the shape/feature hypotheses that are output by the sensing agents; 2) aggregation of the hypotheses at more global level. Note that the candidate shapes/features that are output by the lowest level of the agent hierarchy suffer from high false-positive rate because of complex backgrounds, occlusions, and rather limited fields of view of individual cameras (Figure 1). It is the accumulation of evidence at the higher levels of the hierarchy that progressively eliminates such false positives and provides accurate estimation of human positions in a monitored environment.

Many evidence accumulation schemes have been proposed in the multi camera visual surveillance literature [2], [12], [3], [6], [11], [8], [10], [9]. In [2] and [5], a person's 3D location is estimated by triangulation of 3D rays directed along the line joining camera focal points and the person's centroid in 2D image planes. Bayesian networks have also been used for multi-camera evidence accumulation ([6, 4, 13]). In [6], the Bayesian network fuses independent observations from multiple cameras by iteratively resolving independency relationships and confidence levels within the network. [14] addresses the problem of selecting the best camera position for extracting the desired human motion information. The human position, body orientation and body-side estimation is performed by determining camera viewpoints where these features can be easily estimated and maximizing the joint probabilities of observations obtained from multiple cameras.

Our work reported here takes a different approach to estimating multiple humans, by constitut-

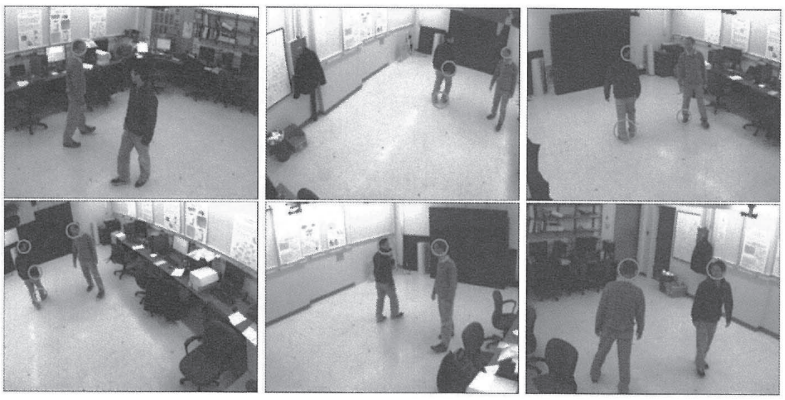


Figure 1: *Images from a multi-camera test sequence with complex background. They were acquired at approximately same time by different cameras. Red circles depict the detected head candidates (both true heads and false positives). Even though there are large number of false positives due to complex background, the proposed evidence accumulation scheme generates accurate 3D head positions.*

ing a framework of agents with heterogeneous characteristics. These agents cooperate to estimate 3D human positions in real time, followed by determination and visualization of their trajectories. This modular agent-based processing architecture makes the proposed framework well-suited to large-scale surveillance applications since new agents can be integrated seamlessly. The evidence accumulation scheme works well even when different cameras are not synchronized in image acquisition times. This paper is an extension of our earlier work on camera networks [7] by more intensively analyzing the system performances through multiple human detection and tracking experiments. In the following sections, we will describe our proposed architecture, and will then show the experimental results to verify the performances.

2 Problem Description

Our overall goal is to develop a cooperative processing architecture for detecting and tracking multiple humans in an environment and visualizing their trajectories. The work presented in this paper solves a sub-problem of the human tracking problem: First to detect the human positions in the environment using individual cameras and, then to effectively combine their information to achieve higher localization accuracy of estimated positions and the reduction of false detections. In solving the detection and localization problem, we have made the following assumptions:

- The environment is defined in terms of the world coordinate frame.
- All cameras are calibrated with respect to the world coordinate frame.
- Image capture in different cameras is not synchronized, although images are acquired with time stamp information.
- Multiple humans may exist in the environment viewed by the network of cameras.

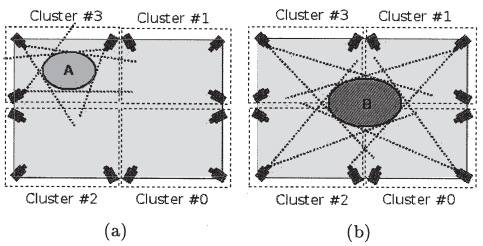


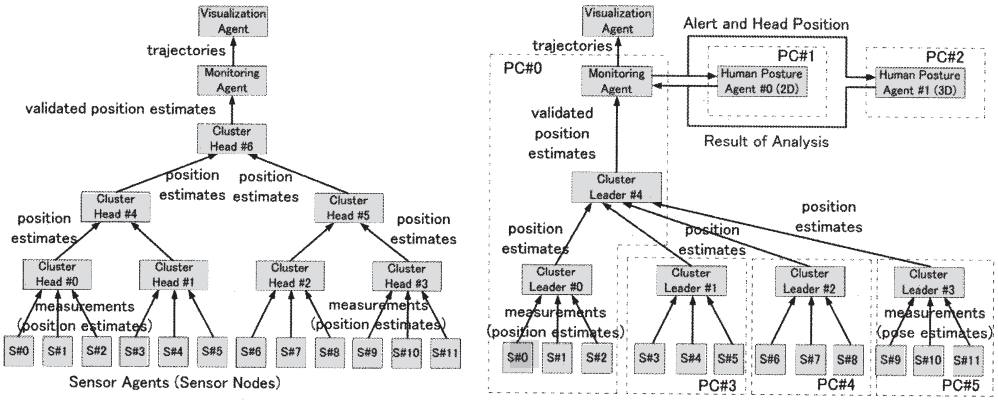
Figure 2: *Camera configuration for evidence accumulation framework: There are 12 cameras grouped into four clusters each monitoring a small part of a rectangular area.*

3 Agent Based Architecture

The cooperative processing architecture consists of the following agents:

- Sensing Agent
- Cluster Leader Agent
- Monitoring Agent
- Visualization Agent

These agents are software processes running on PC's that are connected by wired or wireless network and multiple agents may run on a single PC. The agents



(a) Logical view

(b) an implementation example

Figure 3: *Agent based hierarchical processing architecture for human detection.*

may also control hardware such as cameras for image capture or display devices for visualizing the trajectories of detected humans. Figure 3(a) shows the logical view of an agent based architecture and Figure 3(b) shows an implementation example, explaining how the system is currently set up in our laboratory.

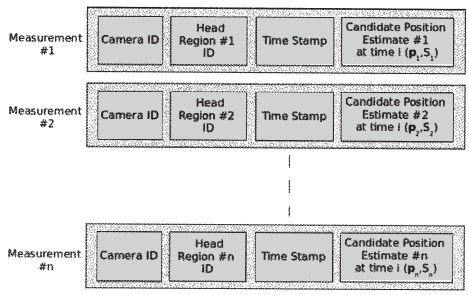


Figure 4: *Data transmitted from a sensing agent to the cluster leader. Cluster leaders also transmits the data in the same format, which ensures a reconfigurable agent architecture.*

3.1 Sensing Agent

The sensing agents are situated at the bottom of agent-based hierarchy. Ideally, in distributed sensor networks, sensor nodes consist of sensors, a processing module and a communication module. In our current setup, we simulate a sensor node by a sensing agent. It is a software agent running on a PC, that utilizes an IEEE 1394 firewire camera for image capture, performs local processing on the acquired images and sends some data to other agents (specifically the cluster leader) at the next higher level of hierarchy. The images are captured with time-stamp information. Local processing involves detecting human-head like object regions (also known as 'head region candidates') in the scene using the output of a background subtraction algorithm. Since sensing always involves false detections, a sensing agent is not expected to always successfully detect the human heads. Its responsibility is only to detect the head region candidates. In the following discussions, we will therefore refer to single camera head region candidates as 'measurements'. The sensing agent then sends a message including the measurements to cluster leader. Figure 4 shows the data structure that the sensing agent sends to a higher level of agents including cluster leaders.

3.2 Cluster Leader Agent

A cluster leader implements our evidence analysis and accumulation algorithm to unify the information received from lower level agents in the agent hierarchy. Note that the hierarchical organization of the agents shown in Figure 3(a) allows for the node below a cluster leader to be either a sensing agent or another cluster leader agent. A cluster leader agent receives messages containing measurements or position estimates from lower level agents and uses them to accumulate evidence for accurate 3D position estimation. There are two types of position estimates:

1. Candidate position estimate (CPE): This is generated in the cluster leader as a result of integrating one or more measurements received from different sensing agents. It is represented by $(\overline{\mathbf{p}}, S)$, where $\overline{\mathbf{p}}$ is the mean vector representing the candidate position and S is the covariance matrix representing position uncertainty. $\overline{\mathbf{p}}$ and S are specified in 3D world coordinates.

2. Validated position estimate (VPE): When a CPE is able to accumulate evidence from three

or more measurements, it is said to be *validated* and is then called as a validated position estimate (VPE). The integration and validation of position estimates is performed using Mahalanobis distances and weighted recursive least squares technique [1]. A VPE is also denoted by $(\overline{\mathbf{p}}, S)$. A CPE may or may not represent an actual human head depending on how many measurements are integrated into it but a VPE represents the position of actual human head.

Once the VPEs are generated, "unnecessary" measurements are eliminated within the cluster leader, to avoid data redundancy and to ensure that each measurement is associated with a unique VPE. The cluster leader then sends message to a higher-level cluster leader to notify the VPEs and also the CPEs which it could not validate. If a cluster leader at the topmost level can not validate any of the CPEs, they are discarded. This cluster leader sends all the VPEs to Monitoring agent and Visualization agent for generating the trajectories and visualization of detected human heads. Note that the cluster leader maintains data structures for keeping track of the CPEs and VPEs. The data structures are exactly the same as those for sensing agents as shown in Figure 4, which therefore ensures the reconfigurability of our cluster leader architecture.

3.3 Monitoring and Visualization Agents

The Monitoring Agent is responsible for monitoring the object/humans found in the environment by associating tracking labels with such objects and also generating trajectories of such objects in motion. The Visualization Agent provides a user interface for visualizing the 3D environment along with the objects/humans found.

3.4 Connectivity and Communication Issues

In a distributed network (wired or wireless), reliability or lack thereof is an important issue. We do not wish to assume a reliable network and we want our framework to allow for fault conditions such as some sensor nodes going down or some communication links failing during a detection and tracking task. To realize an unreliable network, we use the UDP messaging protocol rather than the TCP protocol. A cluster leader integrates the information received from the lower level nodes.

3.5 Configuring Agent Hierarchy

Depending on the number of sensing agents in the camera network, there may be one or more cluster leaders and they may be arranged in multiple layers of the agent hierarchy. There is a trade-off involved between the numbers of levels of the hierarchy in the architecture versus the communication delays in the network. On the one hand, the sensing agents and the cluster leaders may be configured in multiple layers as shown in Figure 2 (a), so that there are multiple clusters of sensing agents and each cluster's data is processed by one cluster leader. Such a configuration will have higher cumulative communication delays compared to a simple network where all the sensing agents are directly connected to a single cluster leader that does all the integration and validation processing. On the other hand, it is typical of wireless sensor networks that the sensing agent nodes may have limited communication range and so may not be able to send their data to a single cluster leader. Therefore, the formation of multiple clusters may be necessary.

If a multiple-cluster formation is allowed, each cluster may be able to cover only a portion of the entire monitored area. In our current system implementation, a cluster leader requires measurements from three or more sensing agents to obtain a VPE. As shown in Figure 2(a), the cluster leader for cluster #3 can validate all the locations within the area A. However, the area B (Figure 2(b)) cannot be covered by sensing agents of any single cluster; so no single cluster leader can validate the locations in this area. The cluster leaders corresponding to all the four clusters need to send their position estimate data to a higher level cluster leader to perform second level of integration. This scenario justifies the need for having multiple levels of cluster leaders in our architecture.

4 Single View Head Detection in Sensing Agent

The human head detection in a single camera image involves contour analysis of foreground silhouettes that may correspond to human objects in the scene. Our algorithm first applies a background subtraction algorithm to a given scene image to obtain foreground silhouettes. It then applies a shape decomposition algorithm [15] to extract head-like circular regions by analyzing the contours of the given silhouettes, as shown in Figure 5. Once we obtain such head-like circular regions (head candidates), we estimate a rough distance d from the camera to head candidate in the 3-D world frame, assuming that a head is modeled by a sphere of radius r in the 3-D world frame as:

$$d = F\,\frac{D}{2r}$$

where F is the focal length of the camera, r the estimated radius of the head region candidate, and

D the diameter of the average human head. This equation indicates that $|\triangle d| \approx (\triangle r/r)\, d$ implying that uncertainty in d is large for a person far away from camera.

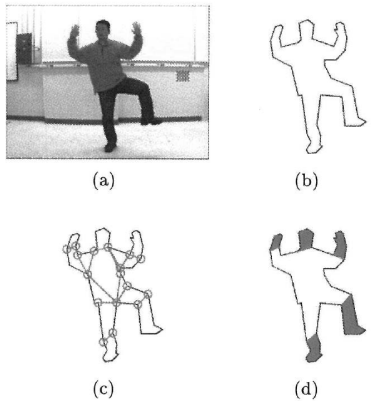


Figure 5: *Zhao's Shape Decomposition (from [15]): (a) the original image, (b) line approximated contour of foreground person, (c) computing the negative curvature minima (represented by small circles) and the cuts (represented by red lines) (d) the edge segments (represented as red colored patches).*

Figure 6 presents an idealized representation of the head candidate detected by a single camera for a single human in its field of view. The head candidate is represented in camera coordinate frame by (u, v, d) where (u, v) are the pixel coordinates of the head candidate region mean and d is its distance from the principal point of the camera. The ellipse in the figure represents uncertainty in d.

The candidate position measurement (u, v, d) obtained from single camera image is transformed to world coordinate frame $\mathbf{p} = (x, y, z)$. Since there is always an uncertainty in (u, v, d) obtained from the camera measurement, the uncertainty involved in (u, v, d) can also be transformed into the 3-D world coordinate frame that is represented by its mean position $\overline{\mathbf{p}}$ and covariance matrix S. For the details of the representation, see our earlier work [7].

The measurement from a single camera may not represent the actual position of a human head. That is why we refer to a detected region as a *head candidate* rather than a head. The reason is that certain non-human objects may appear circular in a single camera view and may be mistaken for a human head. Even if the detected regions actually represent human heads, there is uncertainty in single camera position estimates due to sensor noise and due to assumption about the head size stated previously in this section. This necessitates the evidence accumulation from multiple sensing agents to integrate their measurements to obtain a VPE.

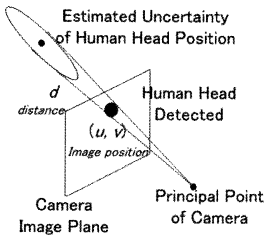


Figure 6: *Single Camera Head Detection.*

5 Multi-Camera Evidence Accumulation in Cluster Leader

When a cluster leader receives a new measurement from a sensing agent, it attempts to update its set of existing position estimates by integrating the new measurement with any one of them. We now describe how this update is carried out using weighted recursive least squares technique with minimum variance.

The human head position in the environment at time t is represented by the position estimate $\mathbf{p} = (\overline{\mathbf{p}}, S)$ where $\overline{\mathbf{p}}$ is the mean vector and S is the covariance matrix representing position uncertainty, as discussed in previous section. Let this position estimate be currently stored in the cluster leader. If a new measurement $\mathbf{p}' = (\overline{\mathbf{p}}', S')$ is received from one of the sensing agents around the same time t, the cluster leader checks to see if this measurement can be integrated with position estimate $\mathbf{p}$, by calculating the Mahalanobis distance between them:

$$d_0 = (\overline{\mathbf{p}} - \overline{\mathbf{p}}')^T (S + S')^{-1} (\overline{\mathbf{p}} - \overline{\mathbf{p}}').$$

If d_0 is less than a certain distance threshold $d_{threshold}$ and if the timestamps of $\mathbf{p}$ and $\mathbf{p}'$ differ by less than a time threshold $T_{threshold}$, they can be integrated. Experimental values of $d_{threshold}$ ranged from 4.0 and 6.0 and $T_{threshold} = 1/7.5$ where the frame rate was 7.5 *fps*. Then $\mathbf{p}$ will be updated, producing the updated estimate $\mathbf{p}_{updated} = (\overline{\mathbf{p}}_{updated}, S_{updated})$ as follows [1]:

1. pre-computation step

$$K = S (S + S')^{-1} \quad (update\ gain)$$

2. update step

$$\begin{aligned} \overline{\mathbf{p}}_{updated} &= \overline{\mathbf{p}} - K (\overline{\mathbf{p}} - \overline{\mathbf{p}}') \\ S_{updated} &= (I - K) S \end{aligned}$$

Since there is a timestamp associated with each position estimate, the time stamp for $\mathbf{p}_{updated}$ is calculated as the average of the time stamps for $\mathbf{p}$ and $\mathbf{p}'$. Integration of one or more measurements results in

CPE and the cluster leader keeps track of how many measurements are integrated into each CPE. In our current implementation, if three or more measurements can be integrated, a CPE becomes validated and is called VPE. Upon validation, all the intermediate CPEs that share any measurement with a VPE are eliminated. This is done primarily to ensure that each measurement only contributes to one VPE in order to minimize false detections. Additionally it leads to efficient memory usage in the cluster leader and fast integration process because there are less number of CPEs and VPEs to integrate any new measurement.

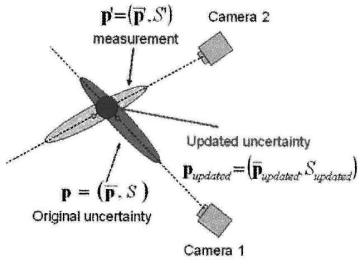


Figure 7: *Uncertainty reduction through measurement integration.*

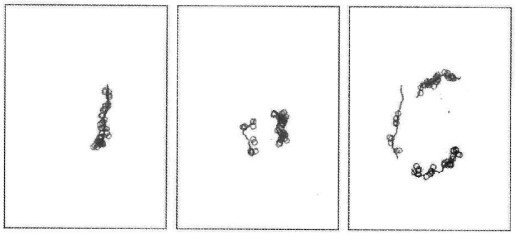
6 Experimental Results

6.1 Evaluation of Localization Accuracy

Our agent-based architecture was implemented using standard PCs (Pentium 4, 3.2 GHz) and 12 cameras (640 × 480 *pixels*, Dragonfly2, Point Grey Research Inc.) whose spatial configuration and network interconnections are shown in Figures 2 and 3(b) respectively, where experiments were made in an indoor rectangular area ($8m \times 5m$). In order to evaluate our system for human head detection, we acquired a video sequence approximately 2 minutes long (frame rate = 7.5 *fps*), where upto three persons are moving around in the rectangular monitoring area. To analyze the head detection performance, three scenarios were considered where either only one, two or three persons were present in the monitoring area. 30 *frames* (time duration = 4 *seconds*) were extracted from the video sequence for each of these scenarios. Therefore in total, we used 90 frames of data which corresponds to 12 second interval.

For comparison purposes, ground truth was generated by manually overlaying circles on human heads in single camera images, which were then integrated to generate ground truth 3D positions using weighted recursive least squares with minimum variance. Since each person was assigned a unique identity in the ground truth data, we generated motion trajectories of the individual persons by linearly interpolating between the ground truth 3D positions. Two different configurations of sensing agents and cluster leaders were considered during the experiments: (a) Configuration 1: A flat structure where all the 12 sensing agents were connected to single cluster leader and (b) Configuration 2: A hierarchical structure where the 12 sensing agents were divided in four clusters of three agents each as shown in Figure 2 and 2 (b). Each of the four clusters has its own cluster leaders and these cluster leaders are connected to a second level cluster leader.

The numerical detection performance of the system is presented in terms of 1) false positive comparison before and after measurement integration and validation in cluster leader, 2) percentage of true positives after measurement integration and validation process and 3) localization accuracy of correctly detected heads.



(a) 1 person (b) 2 persons (c) 3 persons

Figure 8: *Ground truth trajectories and detected head positions reported by system. Black circle represent false positives.*

Table 1: *False positive comparison before/after measurement integration and validation.*

	Configuration 1			Configuration 2		
	before	after	reduction	before	after	reduction
1	19	6	68.4 %	19	11	42.1 %
2	39	14	64.1 %	39	18	53.8 %

Figure 8 graphically illustrates the head detection results for the three scenarios. The solid curves represent the trajectories generated from ground truth positions and circles represent the head detections reported by the system after measurement integration and validation. The black circles denote the false positives. For the two- and three-person scenarios, there are some instances of missed detection. This is because of the complicated background in our test environment which results in many spurious contours in foreground objects, and hence the single camera head detection algorithm performs sub-optimally.

Table 1 presents false positive comparisons before/after measurement integration and validation.

Table 2: *True positive performance after measurement integration and validation in cluster leader.*

	Configuration 1	
	validated heads	% true positives
1	56 (50)	89.3%
2	72 (58)	80.6%
	Configuration 2	
	validated heads	% true positives
1	75 (64)	85.3%
2	85 (67)	78.8%

The reduction in false positives for Configuration 1 is approximately 64-70 % for the three scenarios and for Configuration 2, it ranges from approximately 42-64 %, indicating that there is a significant decrease in the false detections as a result of accumulating evidence from multiple sensing agents. In Table 2, we present the true positive detection performance. The cluster leader integrates the measurements received from sensing agents and generates validated position estimates (VPEs). Not all of these VPEs will be actual human head positions because sometimes false positive 2D measurements may get integrated to give a false positive VPE. But as the high true positive percentages in the table indicate, the system is very effective in filtering out false positive 2D measurements. The reason is that the system needs evidence from at least three sensing agents for generating a VPE. Even if one sensing agent generates a false positive measurement, if it is not corroborated by two other false positive measurements from other sensing agents, it will not satisfy the integration and validation conditions. Table 3 summarizes the mean localization error of the correctly detected heads in world coordinate frame.

From the results, we can observe that both configurations have comparable detection performances. The choice of which configuration to use depends on the extent of monitoring area, scalability, and real time performance. For small monitored areas, few sensing agents are required and therefore we can opt for Configuration 1 due to its simple implementation. In contrast, large monitoring areas with a large number of sensing agents may require Configuration 2. Since the measurement integration and validation process in our system is $O(n)$ for Configuration 1 and $O(log\, n)$ for Configuration 2, the latter configuration is preferable for real time performance. This configuration is also more suited to scalable sensor networks because multiple sensing agents, and cluster leaders can be added in a hierarchical fashion without affecting the performance of other parts of the network. Configuration 2 is also a practical choice for *wireless* sensor networks due to a limited communication range of sensing agent nodes.

Table 3: *Mean localization error in detected head positions in world coordinate frame.*

	Configuration 1	Configuration 2
1	13 cm	15 cm
2	14 cm	14 cm
3	13 cm	13 cm

6.2 Human Tracking and Interaction Experiments

The Monitoring Agent carries out multiple human tracking by associating validated human head estimates obtained in different time instances. This association is achieved by comparing features of validated head estimates in terms of geometric and non-geometric feature attributes such as color attributes (histograms) in the 3D shape and positional proximity in the 3D world frame. Such associations over the time domain generate trajectories of multiple humans in motion. To demonstrate the capability of human tracking in this new *detection-association-based* method, we developed a system called *"Are You Okay?"* where the system interacts with humans in the environment. For example, if a human of interest is in an irregular condition (such as lying down and squatting down suddenly), the Monitoring Agent asks a question like "Are you okay? If you are okay, please raise your right hand." The Monitoring Agent will then investigate the 3-D human posture by communicating with multiple sensing agents to obtain the human contour information. It will then check the gestural response from the human of interest by also communicating 2-D Human Posture Agent or 3-D Human Posture Agent as shown in Figure 2(b). Such demonstrations were successfully performed live in our laboratory. Figure 9 shows a demonstration of multiple human detection and tracking, where two persons are moving around the environment. The top-left window represents the result of human detection in the 3D world coordinate frame in terms of 3D head positions. Figure 10 shows a snapshot of the demonstration "*Are You Okay?*" The Monitoring Agent recognized a human gestural response of raising the right hand, collaborating with a Human Posture Agent as shown in Figure 2 (b). Video clips of such demonstrations are available at

http://cobweb.ecn.purdue.edu/RVL /Research/HumanMotionTracking/index.html.

7 Conclusions

In this paper, we presented a novel evidence accumulation framework for human detection based on agent-based architecture. We conducted experiments to demonstrate good localization accuracy and performances for detecting multiple humans in motion. We also demonstrated a system of tracking and interacting with humans in motion.

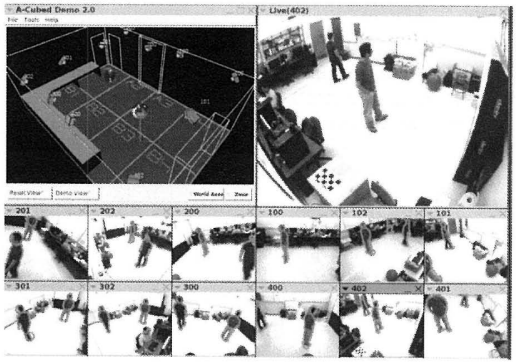


Figure 9: *A demonstration of multiple human detection and tracking, where two persons are moving around the environment. The top-left window shows the result of detection in the 3D world coordinate frame.*

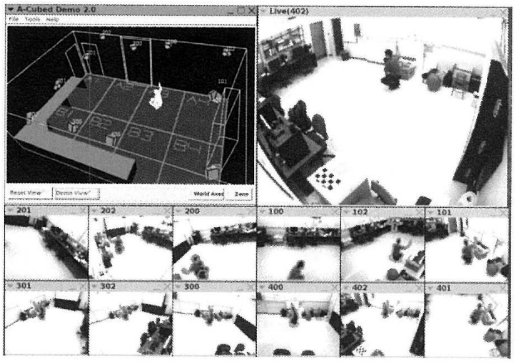


Figure 10: *A snapshot of the demonstration system "Are You Okay" that performs human tracking and interaction with a human in the environment. The Monitoring Agent recognizes a human gestural response of raising the right hand by collaborating with a Human Posture Agent.*

References

[1] Y. Bar-Shalom and Xiao-Rong Li. *Estimation and Tracking: Principles, Techniques and Software.* Artech House, Inc., 1993.

[2] J. Black and T. Ellis. Multi camera image tracking. *Image and Vision Computing*, (24):1256–1267, 2006.

[3] Q. Cai and J.K. Aggarwal. Tracking human motion in structured environments using a distributed-camera system. *IEEE Transactions on Pattern Analysis and Machine Intelligence,*, 21(11):1241–1247, 1999.

[4] T. Chang and S. Gong. Tracking Multiple People with a Multi-Camera System. *Proceedings of IEEE Workshop on Multi-Object Tracking*, page 0019, 2001.

[5] R.T. Collins, O. Amidi, and T. Kanade. An active camera system for acquiring multi-view video. *Proceedings of IEEE International Conference on Image Processing*, 1:I–527–I–520 vol.1, 2002.

[6] S.L. Dockstader and A.M. Tekalp. Multiple camera tracking of interacting and occluded human motion. *Proceedings of the IEEE*, 89(10):1441–1455, Oct 2001.

[7] H. Iwaki, G. Srivastav, A. Kosaka, J. Park, and A. Kak. A novel evidence accumulation framework for robust multi-camera person detection. *Proceedings of ACM/IEEE International Conference on Distributed Smart Cameras*, 2008.

[8] J. Kang, I. Cohen, and G. Medioni. Continuous tracking within and across camera streams. *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 1:I–267–I–272 vol.1, 2003.

[9] V. Kettnaker and R. Zabih. Bayesian multi-camera surveillance. *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2:–259 Vol. 2, 1999.

[10] S. Khan and M. Shah. Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1355–1360, 2003.

[11] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer. Multi-camera multi-person tracking for easyliving. *Proceedings of IEEE International Workshop on Visual Surveillance*, pages 3–10, 2000.

[12] A. Nakazawa, H. Kato, and S. Inokuchi. Human tracking using distributed vision systems. *Proceedings of International Conference on Pattern Recognition*, 1:593–596 vol.1, 1998.

[13] F. Porikli and A. Divakaran. Multi-camera calibration, object tracking and query generation. *Proceedings of IEEE International Conference on Multimedia and Expo*, 1:I–653–6 vol.1, 2003.

[14] A. Utsumi, H. Mori, J. Ohya, and M. Yachida. Multiple-human tracking using multiple cameras. *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pages 498–503, 1998.

[15] L. Zhao. *Dressed Human Modeling, Detection, and Parts Localization.* PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 2001.