Collaborative Sensing via Local Negotiations in Ad Hoc Networks of Smart Cameras

Faisal Z. Qureshi
University of Ontario Institute of Technology
2000 Simcoe Street North, Oshawa, ON, L1H 7K4, Canada
faisal.qureshi@uoit.com

ABSTRACT

The paper develops an *ad hoc* network of active pan/tilt/zoom (PTZ) and passive wide field-of-view (FOV) cameras capable of carrying out observation tasks autonomously. The network is assumed to be uncalibrated, lacks a central controller, and relies upon local decision making at each node and inter-node negotiations for its overall behavior. To this end, we develop intelligent camera nodes (both active and passive) that can perform multiple observation tasks simultaneously. We also present a negotiation protocol that allows cameras nodes to setup collaborative tasks in a purely distributed manner. Camera assignments conflicts that invariably arise in such networks are naturally and gracefully handled through at-node processing and inter-node negotiations. We expect the proposed camera network to be highly scalable due to the lack of any centralized control.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Smart Camera Networks

Keywords

Smart Cameras, Camera Assignment, Collaborative Sensing

1. INTRODUCTION

Future visual sensor networks will rely on smart cameras, which are self-contained vision systems, with increasingly sophisticated image sensors, on-board processing and storage capabilities, power, and (wireless) communication interfaces. Smart cameras networked through wireless links and the Internet provide unprecedented opportunities for developing multi-camera systems capable of visually surveilling extensive public spaces, disaster zones, battlefields, or even entire ecosystems. The effective visual coverage of extensive areas requires multiple cameras to collaborate towards common sensing goals. As the size of these camera networks increase, it becomes infeasible for human operators to monitor the multiple video feeds to identify events of interest,
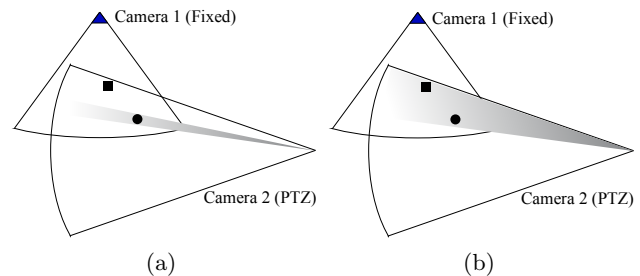
Figure 1: (a) Camera 1 is observing pedestrian A (indicated as a square). Camera 1 attempts to set up a task collaboration with a PTZ camera to acquire a closeup video of this pedestrian. The only PTZ camera in the vicinity (camera 2) is currently observing pedestrian B (indicated as a circle) and it can not meet the requirements set by camera 1. (b) Through negotiations both cameras agree that camera 2 will acquire a lower-quality close-up video of pedestrian A. Consequently, camera 2 is now observing both pedestrians A and B, albeit at a lower-resolution.

or even to control individual cameras directly for the purposes of persistent surveillance. In dynamic environments, especially, camera nodes must be managed in real time to satisfy the observation task requirements. Therefore, it is desirable to design camera sensor networks that are capable of performing visual surveillance tasks autonomously, or at least with minimal human intervention.

In this paper, we demonstrate a camera network model comprising uncalibrated passive and active video cameras that with minimal human assistance can perform persistence surveillance of a large public space. We develop a negotiation framework that allows camera nodes to collaborate with each other in a completely distributed manner to acquire seamless video of pedestrians as they move through a designated region. Camera nodes agree among themselves about how best to carry out the observation tasks through strictly local negotiations. Global consistency, and by extension finding globally optimal solutions, is an unreasonable goal in extremely large scale and rapidly evolving *ad hoc* networks. Therefore, the proposed negotiation model aims to achieve a *satisficing* solution instead of finding the optimal solution. A satisficing solution is often very close to an optimal solution when the cost of obtaining complete information is factored into decision making.

A camera node initiates negotiations in order to establish a collaboration with other cameras in the vicinity. For example, such a collaboration is needed during camera handoffs. The camera that initiates the negotiations is referred to as the client camera; whereas, the cameras in the vicinity that take part in these negotiations are referred to as servers cameras. The client-server metaphor suggests the relationship between the cameras involved in the negotiations. Client cameras set the task description and requirements—e.g., "Acquire closeup video of pedestrian A." Server cameras can either agree to perform the task and meet all its requirements or give a counter offer to the client camera—e.g., "1) Cannot acquire closeup video of pedestrian A. 2) Can observe pedestrian A at a lower resolution."—when these are unable to meet the requirements set by the client camera. Successful negotiations enable cameras to establish mutually acceptable service commitments (Fig. 1). Task collaborations are dynamic arrangements and evolve over time.

We have developed multi-tasking smart camera nodes. These camera nodes are modeled as behavior-based agents and can carry out multiple observation tasks simultaneously. This allows the proposed model to handle camera selection and assignment conflicts in a completely distributed fashion. Additionally, the performance of the camera network degrades gracefully when it is overwhelmed by a multitude of observation tasks.

1.1 Overview

The rest of the paper is organized as follows. Next section discusses the relevant literature and highlights the unique aspects of the work presented here. We discuss the our behavior-based smart camera nodes in Sec. 3. Sec. 4 describes the negotiation model. We present results in Sec. 5 and concludes the paper with discussion and future directions and in Sec. 6.

2. RELATED WORK

Several authors (e.g., [2, 4, 12, 7]) have studied multi-camera issues related to low-level sensing, distributed inference, and tracking. Recently, however, the research community has been paying increasing attention to the problem of controlling or scheduling active cameras in order to capture high-resolution imagery of interesting events. High-resolution imagery not only allows for subsequent biometric analysis, it also helps increase the situational awareness of the surveillance system operators. In a typical setup, information gathered by stationary wide-FOV cameras is used to control one or more active cameras [6, 15, 10]. Generally speaking, the cameras are assumed to be calibrated and the total coverage of the cameras is restricted to the FOV of the stationary camera. Nearly all PTZ scheduling schemes rely on site-wide multi-target, multi-camera tracking. Numerous researchers have proposed camera network calibration to achieve robust object identification and classification from multiple viewpoints, and automatic camera network calibration strategies have been proposed for both stationary and actively controlled camera nodes [14, 5, 3]. Recently, [17] proposes a strategy for proactive PTZ camera control strategy. Here, long-term consequences of camera assignments are taken into account when determining how best to carry out a camera handoff task.

The problems of camera assignment and handoff have mainly been studied in the context of smart camera networks. To perform camera handoffs, [13] construct a distributed lookup
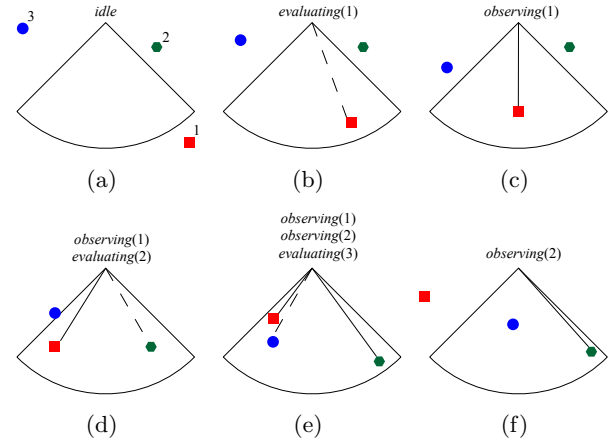
Figure 3: The state of a camera evolves over time. The solid line between the camera and a pedestrian indicates that the camera is observing the pedestrian; where as, a dashed line indicates that the camera is evaluating its suitability to the task of observing that pedestrian. Red square, green hexagon, and blue circle represent pedestrians 1, 2, and 3, respectively.

table, which encodes the suitability of a camera to observe a specific location. For continuous tracking across multiple cameras, [8] propose the use of a handoff function, which is defined as the ratio of co-occurrence to occurrence for point pairs in two views. Their approach does not require calibration or 3D scene information. [11] develop a game theoretic approach to achieve camera handoffs. When a target is visible in multiple cameras, the best camera is selected based on its expected utility. They also propose a number of criteria to construct the utility function, such as the number of pixels occupied by the selected target in an image. Their approach eschews spatial and geometric information. [9] develops a probabilistic framework for selecting the "dominant" camera for observing a pedestrian, defined as the camera with the highest proximity probability, which is computed as the ratio of the foreground blocks occupied by the selected pedestrian and the angular distance between the camera and that pedestrian. [19] presents a game-theoretic strategy for cooperative control of a set of decentralized cameras. The cameras work together to track every target in the area at acceptable image resolutions. The camera network can also be tasked to record higher-resolution imagery of a selected target.

Work by Qureshi and Terzopoulos on *ad hoc* networks of smart cameras [16] is closest to the work presented here. They propose an uncalibrated camera network capable of persistent surveillance. Cameras form groups and cooperatively carry out the observation tasks. Groups are dynamic arrangements and are set up through a distributed auction process. Assignment conflicts are modeled as a constraint satisfaction problem whose solution determines appropriate camera assignments. Their work assume that each camera can only perform a single observation task. We relax this assumption and assume that cameras can engage in more than one observation tasks simultaneously. We show that the resulting network is more robust to observation failures when each camera is allowed to engage in multiple observa-
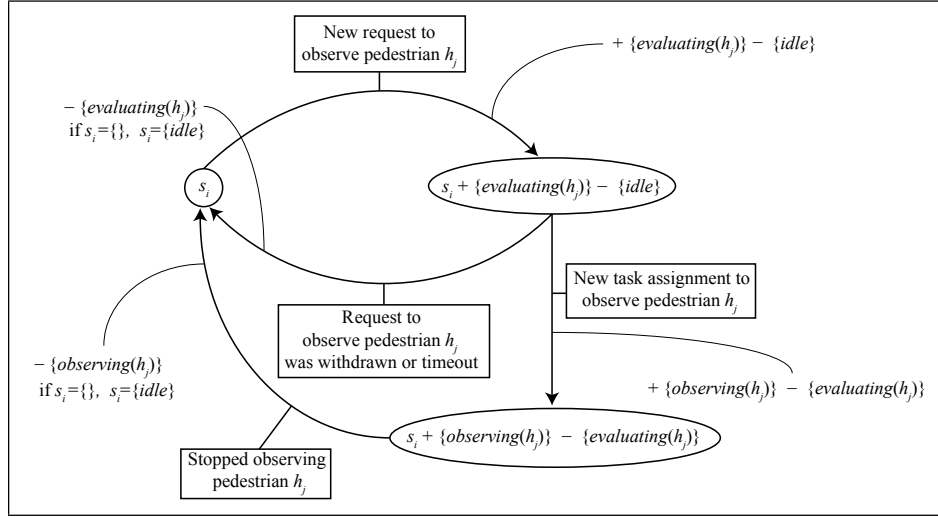
Figure 2: Schema for the infinite state machine. All cameras initially start in the *idle* state. Events are shown in boxes. States are shown in ovals. Each transition adds or removes activities from the state (description) of a camera. Additions and subtractions are shown next to each transition arc. Subtracting a non-existent activity from a state is allowed and does not affect the state.

Level	Behavior	Priority	Camera	Description
0	Sensor	0	Both	Switches on/off the imaging sensor
	Resolution			Selects the resolution of the captured video
1	Search	3		Performs a visual search to acquire a pedestrian given his signature
	Track	2		Tracks a pedestrian
	Fixate		PTZ	Controls the pan/tilt parameters to keep the selected pedestrian(s) within its FOV
	Zoom			Controls the zoom parameter
	Reset	1		Takes PTZ to its default position
2	Evaluate		Both	Identifies a pedestrian and track him for a short duration
	Observe			Tracks a pedestrian

Table 1: Layered behavior routines for passive and active cameras. Behaviors in the higher layers are composed of those found in the lower layers. We assign priorities to Sensor, Search, Track, and Reset behaviors for the purposes of behavior arbitration. A behavior at a higher priority locks out those at the lower priority. E.g., a PTZ camera cannot invoke the Reset behavior when Track behavior is active.

tion tasks simultaneously.

3. SMART CAMERA NODES

Each camera has its own suite of vision routines for pedestrian identification and tracking. We have implemented an appearance-based pedestrian tracker that works reliably across different zoom levels.[1] Each camera has a repertoire of low-level behaviors capable of controlling its sensing operation. Sensing subsystems for both passive and active cameras can be switched off when a camera is not surveilling the scene, say to conserve the battery life. We also assume that our simulated cameras can capture video at two different resolutions: 640×480 or 320×480. Lower resolution video is typically faster to capture and process, 2) requires less space for storage, and 3) have smaller bandwidth requirements for transmission; where as, the higher resolution video is generally more suitable for subsequent biometric analysis.

We treat each camera as a behavior-based autonomous agent.

---
[1]Of course the tracker fails when two pedestrians look alike or when the same pedestrian have a different appearance in two cameras due to changes in viewpoint, lighting, cameras' color responses, etc. Treatment of these issues is not the focus of this work.

The overall behavior of the camera is determined by the vision routines (bottom-up) and the current task (top-down). Camera controller, which is responsible for behavior selection and arbitration, is modeled as an *infinite state machine*. In our case a camera can be in any one of the infinitely many possible states, so a finite state machine is not an appropriate model for our camera controller. The state of a camera represents its current activities. For example, a camera may be tracking one individual and searching for another person at the same time. We begin by listing all the activities (including the null activity representing an idle camera) that a camera can perform:

- $idle(c_i)$: The camera c_i is currently not performing any observation task.

- $observing(c_i, h_j)$: The camera c_i is currently observing pedestrian h_j.

- $evaluating(c_i, h_j)$: The camera c_i is currently evaluating its suitability to observe pedestrian h_j. A camera cannot take part in any negotiations involving h_j without first knowing its suitability to that task; suitability encodes the success probability of a camera with respect to an observation task. Typically, a camera

192
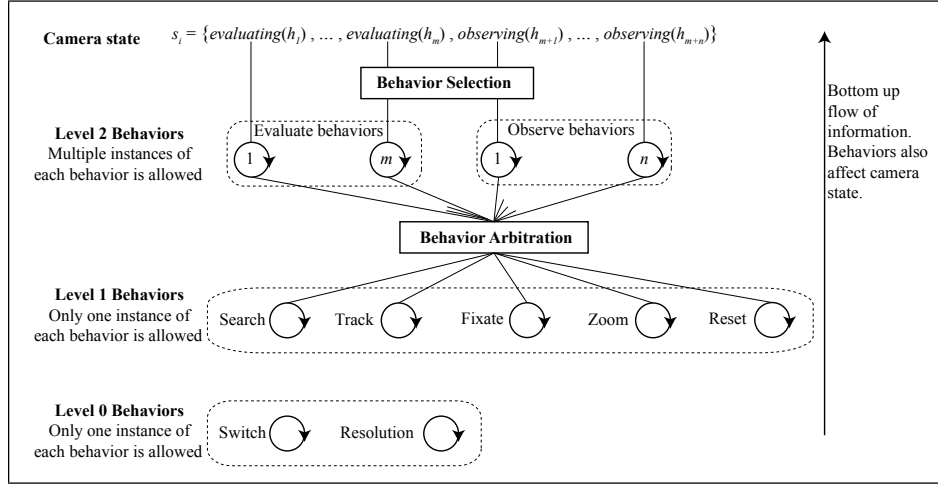
Figure 4: The camera controller is responsible for both behavior selection and arbitration. Fixate, Zoom, and Reset behaviors are only available for active PTZ cameras. Level 0 and 1 behaviors implement a priority based arbitration when priority information is available.

evaluates its suitability to an observation task when it receives the task request from a neighbouring camera.

Given this list of activities, we can define the activity set $\mathcal{A}_i$ for a camera c_i as

$$\mathcal{A}_i = \{idle, evaluating(h_j), observing(h_j)|h_j \in \mathcal{H}\},$$

where $\mathcal{H}$ is the set of pedestrians present in the scene. Notice that the activity set $\mathcal{A}_i$ depends upon the number of individuals $|\mathcal{H}|$ present in the scene and that it is an unbounded set. Let s_i represent the state of camera c_i then $s_i \in \mathcal{S}_i$, where

$$\mathcal{S}_i = \mathcal{P}(\mathcal{A}_i) - \Phi - \mathcal{P}^-(\mathcal{A}_i).$$

$\mathcal{P}(\mathcal{A}_i)$ is the powerset of $\mathcal{A}_i$, Φ represents the empty set, and $\mathcal{P}^-(\mathcal{A}_i)$ consists of logically invalid states. For example, a camera can not be both idle and observing a pedestrian at the same time. Specifically, Let s_i^- denote a logically invalid state for camera c_i then $s_i^- \in \mathcal{P}(\mathcal{A}_i) \cap \mathcal{P}^-(\mathcal{A}_i)$ and either of the following two conditions hold:

Condition 1: $idle(c_i) \in s_i^-$ and $|s_i^-| \geq 2$

Condition 2: $\exists u(h'), v(h'') \in s_i^-$ such that $h' = h''$, where $u, v \in \{evaluating(h_j), observing(h_j)\}$ and $h', h'' \in \mathcal{H}$.

The first condition excludes the states that show a camera as both idle and busy; where as, the second condition excludes the states that show a camera simultaneously observing and evaluating the same individual. Fig. 2 illustrates the rules governing camera state transitions.

Fig. 3(a) depicts an idle camera and three pedestrians. The camera is tasked with observing pedestrian 1. Camera 1 evaluates its suitability to the task of observing pedestrian 1 (Fig. 3(b)) and then accepts the task and starts observing pedestrian 1 (Fig. 3(c)). Next, camera is tasked with observing pedestrian 2. The camera repeats the above process (Fig. 3(d)) and decides to accept the second task as well

(Fig. 3(e)). Meanwhile, camera is asked to observe another pedestrian. The camera deems itself unsuitable for this task and refuses it. The first pedestrian leaves the observational range of the camera and camera is now observing pedestrian 2 (Fig. 3(f)).

Table 1 list behavior routines available to active and passive cameras. We take the layered approach to behavior design, which was first popularized by the *Subsumption* architecture [1]. Here, lower layer behaviors are used to build successively more complex and competent behaviors. Each camera activity is mapped to a corresponding behavior. In our case, camera activities are mapped to level 2 behaviors. These behaviors sit between between discrete activities and continuous operations and presents a well-behaved abstracted view of the underlying continuous reality to the camera controller. Camera controller uses state information, which contains camera activities, to activate/deactivate appropriate behaviors as necessary. Notice that the camera controller does not distinguish between passive wide-FOV and active PTZ cameras, since both expose the same level 2 behaviors: Observe and Evaluate.

3.1 Behavior Arbitration

As is shown in Fig. 4 multiple instances of level 2 behaviors—Observe and Evaluate—may be active simultaneously as there is a one to one mapping between camera activities *observing* and *evaluating* and these two behaviors. All instances of these behaviors, however, rely upon level 1 behaviors: Search, Track, Fixate, Zoom, and Reset (the last three behaviors are only available for active PTZ cameras). Since level 1 behaviors are singletons, instances of behaviors Observe and Evaluate must share level 1 behaviors, which suggests the need for behavior arbitration. Behavior arbitration aims at avoiding negative consequences of behavior interactions—e.g., any one behavior taking over camera resources and locking out all other behaviors, thus leading to observation failures. Behavior arbitration also addresses the problem of *behavior dithering*, where a camera will constantly switch between two (or more) competing tasks.

193
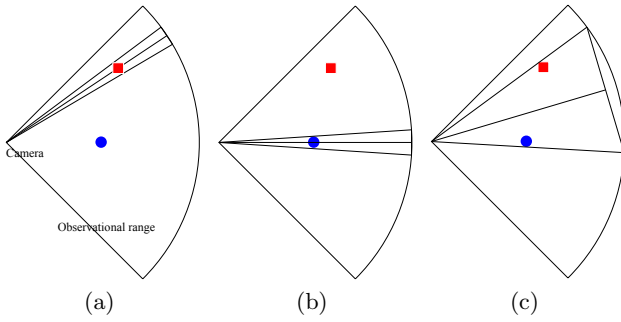
Figure 5: The need for behavior arbitration for active PTZ cameras. (a) Observing pedestrian 1 (indicated as a red square). (b) Observing pedestrian 2 (indicated as a blue circle). (c) Behaviors Observe(Red Square) and Observe(Blue Circle) have struck a compromise. Now the PTZ camera is observing both pedestrians.

3.1.1 Passive Cameras

Behavior arbitration is relatively straightforward in the case of passive cameras. Here, a contention arises when different instances of level 2 behaviors cannot agree on the resolution for video capture. We solve this issue by capturing the video at the highest requested resolution.

3.1.2 Active PTZ Cameras

As shown in Fig. 5, the ability of an active PTZ camera to carry out multiple observation tasks presupposes that competing behaviors are able to work together in an intelligent, goal-driven manner. On the one hand, a PTZ camera can acquire closeup video of a single pedestrian (Fig. 5(a-b)), while on the other hand, it can track multiple pedestrians simultaneously at a much lower resolution by using a wide angle setting (Fig. 5(c)). In our case, a PTZ camera might be carrying out multiple observation tasks simultaneously, so it must be able to intelligently choose between the two extremes: observe a single pedestrian at a higher resolution or observe multiple individuals at lower resolutions.

Fixate and Zoom are image-driven behaviors that strive to maintain an imaginary Region of Interest (ROI) within the field of view. The ROI is defined within the image coordinate system. Specifically, the fixation routine aims to keep ROI in the center of image, while the goal of the zoom routine is to maintain the desired pixel coverage for ROI. When a PTZ camera is acquiring a closeup video of a single pedestrian, ROI is simply the bounding box for that pedestrian as estimated by the pedestrian tracker. For more complex scenarios—e.g., when a PTZ camera is observing multiple pedestrians (Fig. 5(c)), etc.—the ROI is determined by behavior arbitration routine. Fixate and Zoom behaviors operate independently of each other, each trying to achieve its respective goals. Consequently, an active PTZ camera can simultaneously zoom and fixate on the pedestrian(s) being tracked. When a tracking failure is detected, the camera controller goes into a recovery mode where the fixation routine is deactivated and the camera begins to increase its FOV setting with the aim to keep the pedestrian(s) within the field of view. Visual analysis is performed to reacquire the pedestrian using the stored pedestrian signature. If unsuccessful, the camera reports a failure and returns to its default state.

4. NEGOTIATION MODEL

We now explain the negotiation model that allows collaborative tasks in *ad hoc* networks of smart cameras. Task collaborations, which are dynamic arrangements, are established through local negotiations. Consider the following scenario: Camera 1 is currently observing pedestrian A, who is about to exit its observational range. Camera 1 begins negotiations with the neighbouring cameras to take over the task of observing pedestrian A. Camera 1 may also seek a performance commitment from the neighbouring cameras, e.g., acquire closeup video of pedestrian A. Neighbouring cameras are allowed to send in counter offers when they cannot meet the task requirements specified by camera 1. A collaboration is established at the conclusion of a successful negotiation.

Negotiations, which are named and task specific, take place between a client camera node, which wishes to establish a collaboration, and one or more neighbouring cameras, called server camera nodes. Client camera initiates negotiations by sending out a **Msg_New_Task** message to the neighbouring cameras. The message announcing the new observation task also contains task requirements—e.g., frame-rate, resolution, capture closeup video, duration of status messages, etc. Client and server cameras can communicate with each other using **Msg_Server** and **Msg_Client** messages. **Msg_Client** and **Msg_Server** are special messages that can be used for 1) negotiation, 2) task assignment, and 3) task termination. Every camera uses **Msg_Status** message to share its status, including the observation tasks that it is currently performing, to other cameras in the vicinity. Fig. 7 illustrates this process.

It is important to highlight the differences between our approach and other schemes for camera assignment and hand-off presented elsewhere (see [16, 11]). Other schemes assume that a PTZ camera can only carry out a single task at any given time, which is restrictive and often leads to network-level catastrophic observation failures. Consider, for example, the situation shown in Fig. 6. Camera selection and handoff schemes that assume that each camera can only carry out a single observation task simply cannot switch the roles of the two cameras without interrupting one of the observation tasks. Our approach on the other hand can gracefully deals with situations such as these, i.e., where a camera needs to carry out multiple observation task simultaneously.
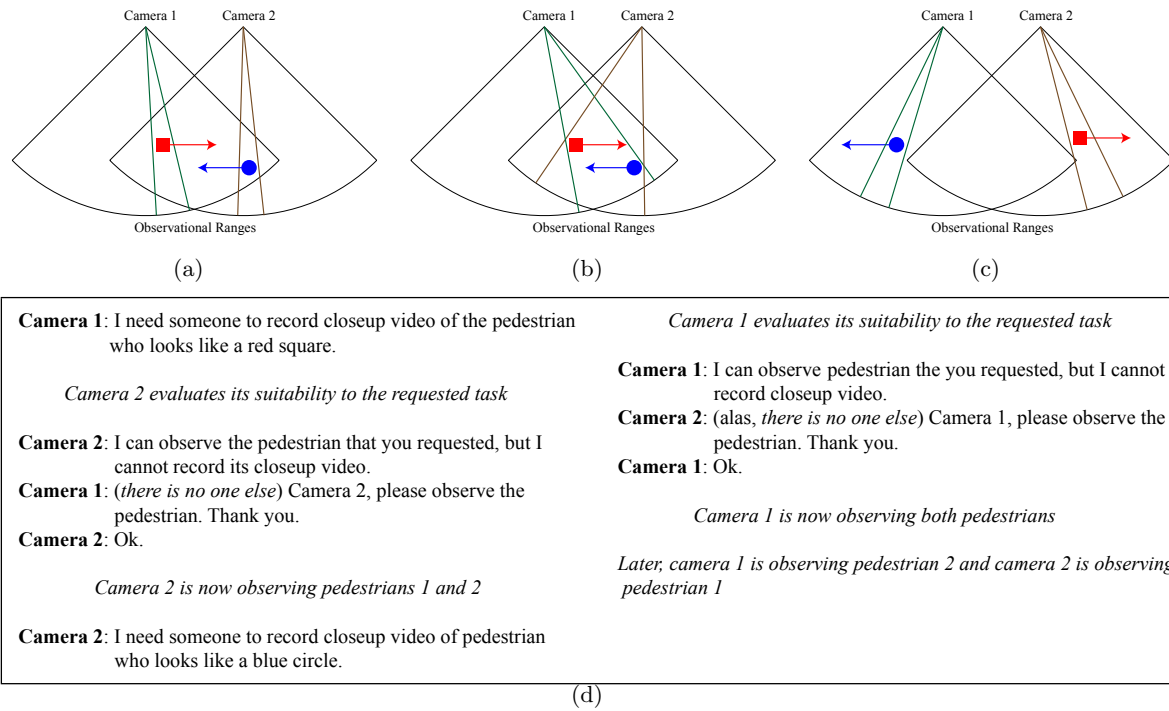
5. RESULTS

To date, we have tested the two aspects of our camera sensor network system: 1) self-managed multi-tasking passive and active camera nodes and 2) collaborative sensing.

5.1 Smart Camera Nodes

Our multi-tasking smart camera nodes are deployed and tested within the Virtual Vision train station simulator [18]. The simulator incorporates a large-scale environmental model (of the original Pennsylvania Station in New York City) with a sophisticated pedestrian animation system that combines behavioral, perceptual, and cognitive human simulation algorithms. The simulator can efficiently synthesize well over 1000 self-animating pedestrians performing a rich variety of activities in the large-scale indoor urban environment. Like real humans, the synthetic pedestrians are fully autonomous. They perceive the virtual environment around them, analyze environmental situations, make decisions and behave

194

(a)   (b)   (c)

Camera 1: I need someone to record closeup video of the pedestrian who looks like a red square.

Camera 2 evaluates its suitability to the requested task

Camera 2: I can observe the pedestrian that you requested, but I cannot record its closeup video.
Camera 1: (*there is no one else*) Camera 2, please observe the pedestrian. Thank you.
Camera 2: Ok.

Camera 2 is now observing pedestrians 1 and 2

Camera 2: I need someone to record closeup video of pedestrian who looks like a blue circle.

Camera 1 evaluates its suitability to the requested task

Camera 1: I can observe pedestrian the you requested, but I cannot record closeup video.
Camera 2: (alas, *there is no one else*) Camera 1, please observe the pedestrian. Thank you.
Camera 1: Ok.

Camera 1 is now observing both pedestrians

*Later, camera 1 is observing pedestrian 2 and camera 2 is observing pedestrian 1*

(d)

Figure 6: (a) Camera 1 is an active PTZ camera that is acquiring closeup video of pedestrian 1 (indicated as a red square). Camera 2 is also an active PTZ camera that is acquiring closeup video of pedestrian 2 (indicated as a blue circle). Pedestrian 1 will soon leave the observational range of camera 1 and enter the observational range of camera 2. Pedestrian 2 on the other hand is entering the observational range of camera 1. Camera 1 and 2 must switch their roles—camera 1 should start observing pedestrian 2 and camera 2 should start observing pedestrian 1. (b) Through negotiations both cameras decide to observe the two pedestrians simultaneously. (c) Camera 2 is now acquiring closeup video of pedestrian 1, while camera 1 is busy acquiring closeup video of pedestrian 2. (d) Hypothetical negotiations between the two cameras. For the first half of the negotiations camera 1 acts as the client and camera 2 acts as a server; whereas, later during negotiations involving pedestrian 2 (indicated as a blue circle) camera 2 acts as a client and initiates the negotiations and camera 1 acts as the server.

naturally within the train station. They can enter the station, avoiding collisions when proceeding though portals and congested areas, queue in lines as necessary, purchase train tickets at the ticket booths in the main waiting room, sit on benches when they are tired, purchase food/drinks from vending machines when they are hungry/thirsty, etc., and eventually proceed downstairs in the concourse area to the train tracks. Standard computer graphics techniques enable a photorealistic rendering of the busy urban scene with considerable geometric and photometric detail (Fig. 8).

We have implemented an interface that presents the operator a display of the synthetic video feeds from multiple virtual surveillance cameras. The operator can select a person in any video feed and instruct the camera network to perform one of the following tasks: 1) follow the person, 2) capture a high-resolution snapshot, or 3) zoom-in and follow the person. The network then automatically assigns cameras to fulfill the task requirements. The operator has the option of assigning multiple tasks to the same camera. Our initial results are promising. Cameras successfully carried out multiple observation tasks simultaneously. Some of the problems that we encountered are related to pedestrian identification and tracking failures.
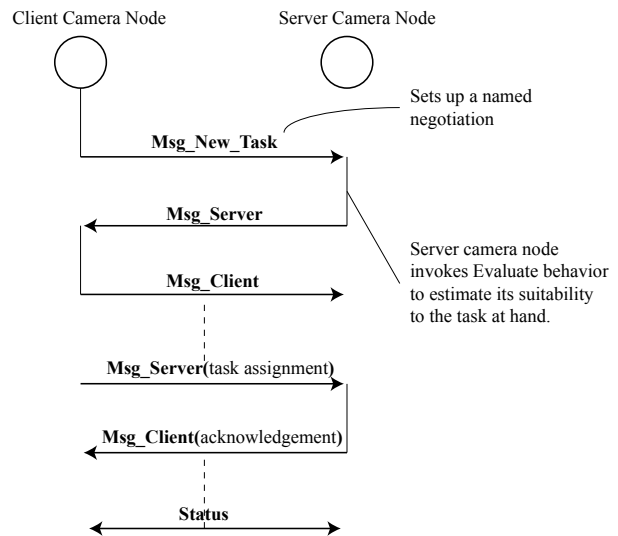
Figure 7: A client camera node initiates a negotiation session with a server camera.
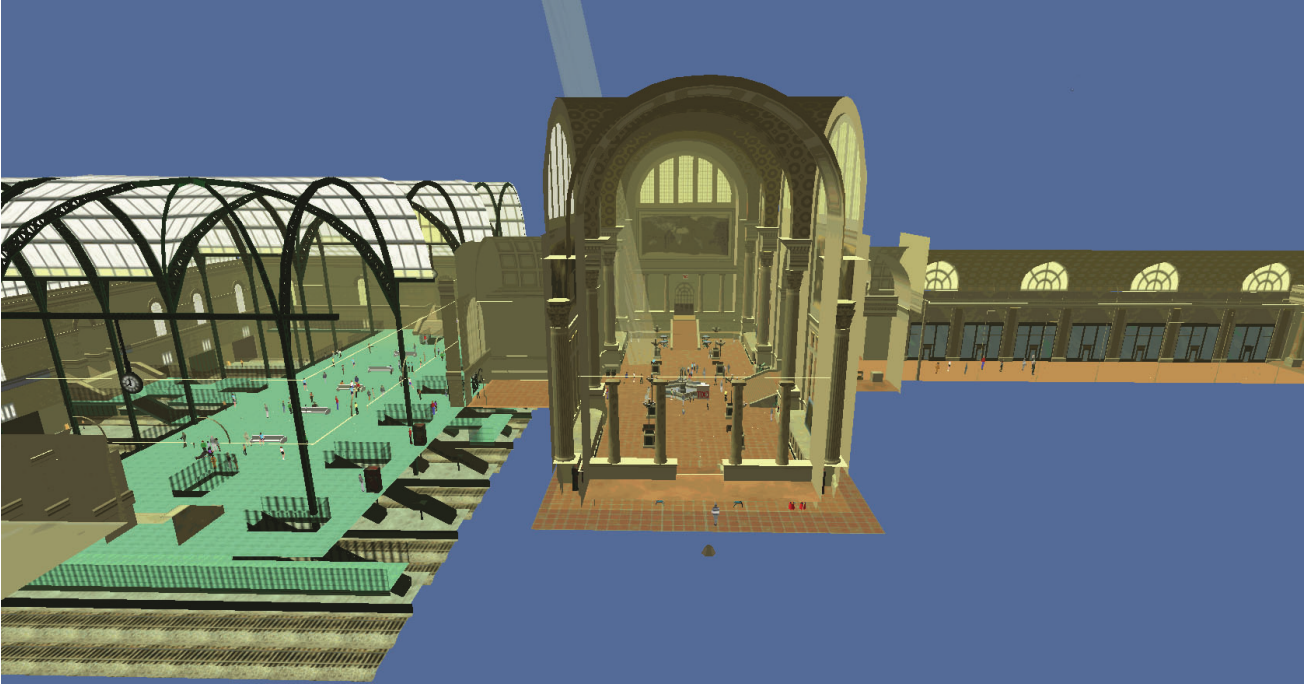
Figure 8: A cutaway side view of the virtual train station. The main waiting room is at the center, the arcade at the right, and the concourses at the left with stairs leading down to the train platforms and tracks. The train station is populated by autonomous, self-animating virtual pedestrians.

Number of pedestrians	Average resolution video of the captured video
1	293×54 pixels
2	200×44 pixels
4	98×13 pixels
8	54×5 pixels

Table 2: Average resolution of the captured video with two active PTZ cameras. Cameras' frame resolution is 640×480 pixels.

5.2 Collaborative Sensing Task

First, we only consider two cameras (1 & 2) in the network shown in Fig. 9. The task is to acquire closeup video of the selected pedestrians as the pass through the waiting room on their way to the arcade. Cameras 1 and 2 need to perform handoff when a pedestrian leaves the waiting room and enters the arcade. We run the scenario with 1, 2, 4, and 8 pedestrians. These pedestrians start at different locations in the waiting room at the same time and walk towards the arcade. We repeated each run twice and recorded the quality of the closeup video captured for each pedestrian in each case. Table 2 list the average quality of the captured video for 1, 2, 4, and 8 pedestrians. Notice that video capture quality continues to decrease as the number of pedestrians that need to be observed simultaneously increase. PTZ cameras are behaving as passive wide-FOV cameras when 8 pedestrians are present in the scene in order to observe all of them simultaneously. This suggests that performance degrades gracefully when we increase the number of simultaneous observation tasks.

Next, we choose a switching task where two camera are asked to perform simultaneous handoffs without interrupting the observation task. Camera 1 is initially observing a pedestrian who is in the waiting room and is moving towards the arcade, while camera 2 is observing a pedestrian who is in the arcade and who wishes to go into the waiting room. We repeated this experiment 5 times with different initial conditions. The camera network successfully performed the switching task every time. It is fair to say that without multi-tasking two PTZ cameras cannot perform switching without interrupting one of the observation tasks.

Lastly, we tasked our camera network comprising simulated active and passive cameras (Fig. 9) with observing up to 16 pedestrians. The camera network consisted of 3 active PTZ cameras and 1 passive wide-FOV camera. The goal was to "simply" observe all the pedestrians during their entire stay within the observational ranges of these four cameras. We noticed that very quickly the 3 PTZ cameras settled into a "passive camera behavior." All 16 pedestrians were successfully observed at all times.

6. DISCUSSION AND CONCLUSIONS

Future vision surveillance systems will be *ad hoc* networks of stationary and active cameras capable of providing perceptive coverage of extended environments with minimal, if any, reliance on a human operator. In addition to robust, low-level vision routines, such systems will also require novel camera network methodologies. The work presented in this paper is another step toward the realization of smart camera networks and our initial results appear promising. The overall behavior of our network is governed by local decision making at each camera node and negotiations between neighbouring camera nodes.

(a) Active PTZ camera   (b) Active PTZ camera   (c) Active PTZ camera   (d) Passive wide-FOV camera
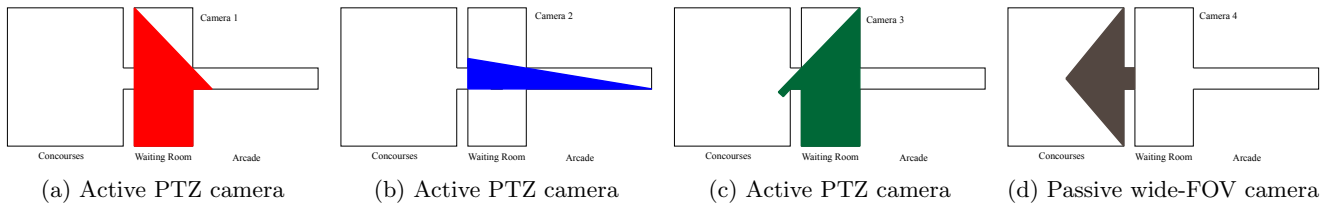
Figure 9: Our four camera simulated network deployed in the virtual train station (Fig. 8). Observational ranges for different cameras are shown as shaded regions.

The proposed network carries out collaborative observation tasks in a completely distributed manner, which suggests that the proposed approach is well-suited to large-scale camera networks. Our approach shuns camera calibration or a detailed world model, which makes it suitable for *ad hoc* networks of smart cameras. Prior camera network calibration, we believe, is an unrealistic goal for a large-scale camera network consisting of heterogeneous cameras. Similarly, our approach does not expect a detailed world model which, in general, is hard to acquire.

Specifically, we have developed behavior-based smart camera nodes capable of carrying out multiple observation tasks simultaneously. We have also proposed a negotiation model for setting up collaborative sensing tasks in a distributed manner. Furthermore, the performance of the camera network degrades gracefully when it is overwhelmed by a multitude of observation tasks. In the future, we plan to evaluate the proposed approach on larger simulated and physical camera networks.

7. ACHNOWLEDGEMENTS

8. REFERENCES

[1] R. A. Brooks. A Robust Layered Control System for a Mobile Robot. In *IEEE Journal of Robotics and Automation*, volume RA-2 (1), Apr. 1986.

[2] R. Collins, O. Amidi, and T. Kanade. An Active Camera System for Acquiring Multi-View Video. In *Proc. International Conference on Image Processing (ICIP02)*, pages 517–520, Rochester, NY, Sept. 2002.

[3] D. Devarajan, R. J. Radke, and H. Chung. Distributed metric calibration of ad hoc camera networks. *ACM Transactions on Sensor Networks*, 2(3):380–403, Aug. 2006.

[4] R. Farrell and L. S. Davis. Decentralized Discovery of Camera Network Topology. In *Proc. of the Second International Conference on Distributed Smart Cameras (ICDSC08)*, pages 1–8, Menlo Park, CA, Sept. 2008.

[5] T. Gandhi and M. M. Trivedi. Calibration of a reconfigurable array of omnidirectional cameras using a moving person. In *Proc. ACM International Workshop on Video Surveillance and Sensor Networks (VSSN04)*, pages 12–19, New York, NY, 2004. ACM Press.

[6] A. Hampapur, S. Pankanti, A. Senior, Y.-L. Tian, L. Brown, and R. Bolle. Face cataloger: Multi-scale imaging for relating identity to location. In *Proc. IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS03)*, pages 13–21, Washington, DC, 2003.

[7] K. Heath and L. Guibas. Multi-person tracking from sparse 3D trajectories in a camera sensor network. In *Proc. of the Second International Conference on Distributed Smart Cameras (ICDSC08)*, pages 1–8, Menlo Park, CA, Sept.

2008.

[8] Y. Jo and J. Han. A new approach to camera hand-off without camera calibration for the general scene with non-planar ground. In *Proc. of the 4th ACM international workshop on Video surveillance and sensor networks (VSSN06)*, pages 195–202, Santa Barbara, CA, Oct. 2006. ACM.

[9] J. Kim and D. Kim. Probabilistic camera hand-off for visual surveillance. In *Proc. Second ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC08)*, pages 1–8, Stanford, CA, Sept. 2008.

[10] N. O. Krahnstoever, T. Yu, S. N. Lim, K. Patwardhan, and P. H. Tu. Collaborative Real-Time Control of Active Cameras in Large-Scale Surveillance Systems. In *Proc. ECCV Workshop on Multi-camera and Multi-modal Sensor Fusion*, pages 1–12, Marseille, France, Oct. 2008.

[11] Y. Li and B. Bhanu. Utility-based Dynamic Camera Assignment and Hand-off in a Video Network. In *Proc. of the Second International Conference on Distributed Smart Cameras (ICDSC08)*, pages 1–9, Menlo Park, CA, Sept. 2008.

[12] P. B. L. Meijer, C. Leistner, and A. Martiniere. Multiple View Camera Calibration for Localization. In *Proc. of the First IEEE/ACM International Conference on Distributed Smart Cameras (ICDSC07)*, pages 228–234, Vienna, Austria, Sept. 2007.

[13] J. Park, P. C. Bhat, and A. C. Kak. A Look-up Table Based Approach for Solving the Camera Selection Problem in Large Camera Networks. In B. Rinner and W. Wolf, editors, *Working Notes of the International Workshop on Distributed Smart Cameras (DSC06)*, pages 72–76, Boulder, CO, Oct. 2006.

[14] F. Pedersini, A. Sarti, and S. Tubaro. Accurate and simple geometric calibration of multi-camera systems. *Signal Processing*, 77(3):309–334, Mar. 1999.

[15] F. Z. Qureshi and D. Terzopoulos. Surveillance Camera Scheduling: A Virtual Vision Approach. *ACM Multimedia Systems Journal*, 12(3):269–283, Dec. 2006.

[16] F. Z. Qureshi and D. Terzopoulos. Smart Camera Networks in Virtual Reality. *Proceedings of the IEEE (Special Issue on Smart Cameras)*, 96(10):1640–1656, Oct. 2008.

[17] F. Z. Qureshi and D. Terzopoulos. Planning Ahead for PTZ Camera Assignment and Control. In *Proc. Third ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 09)*, pages 1–8, Como, Italy, Aug. 2009.

[18] W. Shao and D. Terzopoulos. Autonomous Pedestrians. *Graphical Models*, 69(5-6):246–274, 2007.

[19] B. Song, C. Soto, A. K. Roy-Chowdhury, and J. A. Farrell. Decentralized camera network control using game theory. In *Proc. of the Second IEEE/ACM International Conference on Distributed Smart Camers (ICDSC08)*, pages 1–8, Menlo Park, CA, Sept. 2008.