Chapter 1

PROACTIVE PTZ CAMERA CONTROL

A Cognitive Sensor Network That Plans Ahead

Faisal Z. Qureshi
Faculty of Science, University of Ontario Institute of Technology
faisal.qureshi@uoit.ca

Demetri Terzopoulos
Computer Science Department, University of California, Los Angeles
dt@cs.ucla.edu

Abstract We present a visual sensor network—comprising wide field-of-view (FOV) passive cameras and pan/tilt/zoom (PTZ) active cameras—capable of automatically capturing closeup video of selected pedestrians in a designated area. The passive cameras can track multiple pedestrians simultaneously and any PTZ camera can observe a single pedestrian at a time. We propose a strategy for proactive PTZ camera control where cameras plan ahead to select optimal camera assignment and handoff with respect to predefined observational goals. The passive cameras supply tracking information that is used to control the PTZ cameras.

Keywords: Smart cameras, camera networks, computer vision, PTZ cameras, visual surveillance, persistent human observation

1. Introduction

Automated human surveillance systems comprising fixed CCTV cameras can detect and track multiple people, but they perform poorly on tasks that require higher resolution images, such as acquiring closeup facial images for biometric identification. On the other hand, active pan/tilt/zoom (PTZ) cameras can be used to capture high-quality video of relevant activities in the scene. This has led to surveillance systems that combine passive wide field-of-view (FOV) cameras and active PTZ

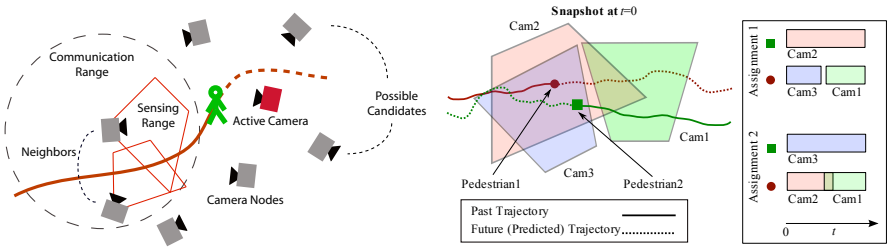


Figure 1.1: (a) A camera network for video surveillance consists of camera nodes that can communicate with other nearby nodes. Collaborative, persistent surveillance requires that cameras organize themselves to perform camera handover when the observed subject moves out of the sensing range of one camera and into that of another. (b) The need for planning in camera assignment and handoff: A control strategy that does not reason about the long-term consequences of camera assignments might prefer Assignment 1 over Assignment 2, which would eventually lead to an observation failure.

cameras. Typically, the PTZ camera control routines rely on the tracking information supplied by the passive cameras.

Manual control of PTZ cameras is clearly infeasible for large networks, especially as the number of persons and activities in the scene exceeds the number of available PTZ cameras. Consequently, it is desirable to develop control strategies that enable the PTZ cameras to carry out observation tasks autonomously or with minimal human intervention. The dynamic nature of the observation task greatly complicates the problem of assigning cameras to observe different pedestrians.

We tackle the challenging problem of controlling active PTZ cameras in order to capture seamless closeup video of pedestrians present in a designated area. In general, no single camera is able to achieve this goal, as the pedestrians enter and exit the observational ranges of different cameras (Fig. 1.1(a)). Furthermore the camera network must be able to resolve conflicts that might arise when tasking multiple cameras to observe different pedestrians simultaneously. We treat the control of active PTZ cameras as a *planning* problem whose solution achieves optimal camera utilization with respect to predefined observational goals. A successful camera assignment and handoff strategy should consider both the short-term and long-term consequences of camera assignments when deciding how to carry out an observational task.

Consider, for example, the scenario shown in Fig. 1.1(b). A control strategy that does not *reason* about the long-term consequences of camera assignment might task Cam3 to observe the red pedestrian (indicated

by the circle) and task Cam2 to observe the green pedestrian (indicated by the square). While this assignment may satisfy the immediate goals, it creates a problem as the green pedestrian continues moving to the right while the red pedestrian continues moving to the left. It is impossible to capture seamless closeup video of the red pedestrian as he moves from Cam3 to Cam1 since there is no overlap between the two cameras. On the other hand, a camera control strategy that reasons about the long-term consequences of camera assignment should assign Cam2 to the red pedestrian and Cam3 to the green pedestrian, assuming that both pedestrians will continue moving in their current directions. This allows for a seamless handoff between Cam2 and Cam3.

The type of research that we report here would be very difficult to carry out in the real world given the expense of deploying and experimenting with an appropriately complex smart camera network in a large public space such as an airport or a train station. Moreover, privacy laws generally restrict the monitoring of people in public spaces for experimental purposes. To bypass the legal and cost impediments, we espouse Virtual Vision, a unique synthesis of computer graphics, artificial life, and computer vision technologies [Qureshi and Terzopoulos, 2008]. Virtual Vision is an advanced simulation framework for working with machine vision systems, including smart camera networks, that also offers wonderful rapid prototyping opportunities. Exploiting visually and behaviorally realistic environments, called reality emulators, virtual vision offers significantly greater flexibility and repeatability during the camera network design and evaluation cycle, thus expediting the scientific method and system engineering process. Our companion chapter in this volume provides a more detailed review of the Virtual Vision paradigm.

Related Work

Several authors (e.g., [Collins et al., 2002; Farrell and Davis, 2008; Meijer et al., 2007; Heath and Guibas, 2008]) have studied multicamera 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 [Hampapur et al., 2003; Qureshi and Terzopoulos, 2006; Krahnstoever et al., 2008]. 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 multitarget, multicamera tracking. Numerous researchers have proposed camera network calibration to achieve robust object identification and classifi-

cation from multiple viewpoints, and automatic camera network calibration strategies have been proposed for both stationary and actively controlled camera nodes [Pedersini et al., 1999; Gandhi and Trivedi, 2004; Devarajan et al., 2006].

The problems of camera assignment and handoff have mainly been studied in the context of smart camera networks. To perform camera handoffs, [Park et al., 2006] construct a distributed lookup table, which encodes the suitability of a camera to observe a specific location. For continuous tracking across multiple cameras, [Jo and Han, 2006] 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. [Li and Bhanu, 2008] 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. [Kim and Kim, 2008] develop 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. [Song et al., 2008] present 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.

Our work on proactive camera control differs from prior work in an important way. With the notable exception of [Krahnstoever et al., 2008], existing schemes for camera assignment do not *reason* about the long-term consequences of camera assignments and handoffs. In essence, existing schemes are purely reactive. By contrast, the strategy introduced in this paper is *proactive* and deliberative. When searching for the best current camera assignment, it considers the future consequences of possible camera assignments. The ability to reason about the future enables our system to avoid camera assignments that might appear optimal at present, but will eventually lead to observation and tracking failures. For an overview of planning and search techniques, we refer the reader to [Russell and Norvig, 2003].

2. Proactive Camera Control

Problem Statement

Consider a camera network comprising N_p calibrated wide FOV passive cameras and N_a PTZ active cameras. The passive cameras track

and estimate the 3D positions and velocities of the observed pedestrians. Let $\mathcal{H} = \{h_j | j = 1, 2, \cdots\}$ denote the set of pedestrians observed during the operation of the camera network. At time instant t, the state of the pedestrians observed by the camera network is given by $(\mathbf{x}_i^t, \mathbf{v}_i^t)$, where $\mathbf{x}_i$ and $\mathbf{v}_i$ represent the ground plane position and velocity, respectively, of observed pedestrian i. Let $C = \{c_i | i \in [1, N_a]\}$ denote the set of active PTZ cameras. Each PTZ camera is described by a tuple $\langle \mathbf{o}, \alpha_{\min}, \alpha_{\max}, \beta_{\min}, \beta_{\max} \rangle$, where we assume that the 3D position $\mathbf{o}$ of each PTZ camera is known *a priori*, and where $[\alpha_{\min}, \alpha_{\max}]$ and $[\beta_{\min}, \beta_{\max}]$ represent pan and tilt limits, respectively, for each PTZ camera. Furthermore, we assume that each PTZ camera stores a map between the gaze direction parameters (α, β) and 3D world locations. In [Qureshi and Terzopoulos, 2006], we describe how such a map can be automatically learned by observing pedestrians present in the scene. Thus, given the 3D location of the pedestrian, a PTZ camera is able to direct its gaze towards the pedestrian. We model each PTZ camera as an autonomous agent—complete with search, fixate, and zoom behaviors and low-level pedestrian tracking routines—that is capable of recording closeup video of a designated pedestrian without relying on continuous feedback from passive cameras.

With the above assumptions, we formulate collaborative camera control as a centralized planning problem. We favor a centralized planner as it is not obvious how to cast the problem of capturing closeup video of the selected pedestrians within a distributed planning framework. However, we remain cognizant of the fact that centralized planning is not suitable for large networks of PTZ cameras due to *timeliness* concerns.[1]

A planning problem is characterized by states, actions, and goals, and its solution requires finding a sequence of actions that will take an agent from its current state to a goal state. Fig. 1.2 defines the states, actions, goal, etc., for our system. We can then formulate the solution state sequence to the planning problem as

$$\mathcal{S}^* = \underset{\mathcal{S} \in \mathcal{S}_a}{\operatorname{argmax}} \, \mathcal{Q}(\mathcal{S}),$$

where $\mathcal{Q}(\mathcal{S})$ is the quality of state sequence $\mathcal{S}$ among a set of admissible state sequences $\mathcal{S}_a$. The corresponding action sequence is $\mathcal{A}^*$.

Finding Good State Sequences

The overall performance of the camera network is intimately tied to how capable the individual PTZ cameras are at carrying out the observation tasks assigned to them. In order for the planner to find the plan with the highest probability of success, we must quantify the quality of a state sequence (or a plan) in terms of the expected performance of individual PTZ cameras. We construct a probabilistic objective function that describes the quality of a state sequence in terms of the success

Definition 1 (State) Tuple s^t represents the state of the system during time interval $[t, t+1)$. $s^t = \langle s_i^t | i = 1, \cdots, N_a \rangle$, where s_i^t denotes the status of the PTZ camera i at time t. Possible values for s_i^t are $\mathit{Free}(c_i)$, $\mathit{Acquiring}(c_i, h_j)$, or $\mathit{Recording}(c_i, h_j)$, for pedestrian $h_j \in \mathcal{H}$ and for camera $c_i \in C$.

Definition 2 (Actions) Each PTZ camera has a repertoire of four actions: *Acquire*, *Record*, *Continue*, and *Idle*. Table 1.1 tabulates these actions, along with their preconditions and effects. a_i^t denotes the action for PTZ camera c_i at time t. The *Continue* action instructs a PTZ camera to continue its current behavior.

Definition 3 (Joint Action) At any given instant, each PTZ camera is executing exactly one action (possibly a *Continue*). The concurrent set of actions across the different cameras is called a *joint action*. The tuple $a^t = \langle a_i^t | i = 1, \cdots, N_a \rangle$ represents the joint action of N_a PTZ cameras at time t.

Definition 4 (Action Sequence) Let $\mathcal{A} = \{a^t | t = 0, 1, \cdots\}$ denote an action sequence. Note that the elements of an action sequence are joint actions.

Definition 5 (State Sequence) Let $\mathcal{S} = \{s^t | t = 0, 1, \cdots\}$ denote a state sequence. Sequence $\mathcal{S}$ is obtained by starting in some initial state s^0 and applying an action sequence $\mathcal{A}$. We express the quality of the sequence $\mathcal{S}$ as $\mathcal{Q}(\mathcal{S})$, which we define in the next section.

Definition 6 (Goal) The goal of the system is to capture closeup video of selected pedestrians during their presence in a designated area. Our choice of goal leads to the notion of *admissible* state sequences. An admissible state sequence satisfies the observational constraints. Consider, for example, the goal of observing pedestrians $h \subset \mathcal{H}$ during the time interval $[t_s, t_e]$. Then, $\mathcal{S}_a = \{s^t | t = t_s, \cdots, t_e\}$ represents an admissible state sequence if $(\forall t \in [t_s, t_e])(\exists i \in [1, N_a]) s_i = \mathit{Acquiring}(c_i, h_j) \vee \mathit{Recording}(c_i, h_j)$, where $h_j \in h$ and $c_i \in C$. Clearly, our notion of an admissible state sequence must be revised to cope with situations where the FOVs of the cameras do not overlap.

Figure 1.2: States, actions, and goal of our planning problem.

Actions	Preconditions	Effects	Description
$\mathit{Continue}(c_i)$	none	none	Do nothing
$\mathit{Idle}(c_i)$	none	$s_i = \mathit{Free}(c_i)$	Stop recording
$\mathit{Acquire}(c_i, h_j)$	$s_i \neq \mathit{Acquiring}(c_i, h_j) \wedge s_i \neq \mathit{Recording}(c_i, h_j)$	$s_i = \mathit{Acquiring}(c_i, h_j)$	Start recording pedestrian h_j
$\mathit{Record}(c_i, h_j)$	$s_i = \mathit{Acquiring}(c_i, h_j)$	$s_i = \mathit{Recording}(c_i, h_j)$	Keep recording pedestrian h_j

Table 1.1: Action schema for PTZ cameras ($c_i, i \in [1, n]$).

probabilities of the individual PTZ cameras. Such an objective function then enables the planner to compute the plan that has the highest probability of achieving the goals.

PTZ Camera Relevance. We begin by formulating the relevance $r(c_i, O)$ of a PTZ camera c_i to an observation task O. The relevance encodes our expectation of how successful a PTZ camera will be at satisfying a particular observation task; i.e.,

$$p(c_i|O) = r(c_i, O),$$

Camera-pedestrian distance r_d: gives preference to cameras that are closer to the pedestrian.

Frontal viewing direction r_γ: gives preference to cameras having a frontal view of the pedestrian.

PTZ limits $r_{\alpha\beta\theta}$: takes into account the turn and zoom limits of the PTZ camera.

Observational range r_o: reflects the observational constraints of a camera. It is set to 0 when the pedestrian is outside the observational range of a camera; otherwise, it is set to 1.

Handoff success probability r_h: gives preference to handoff candidates in the vicinity of the camera currently observing the pedestrian. The idea is that nearby cameras have a similar viewpoint, making the appearance-based pedestrian signature more relevant for the candidate camera. Factor r_h is considered only during camera handoffs; otherwise, it is set to 1. A consequence of using this factor in the camera relevance computation is that the planner will prefer plans with fewer handoffs, which is desirable.

Figure 1.3: These five factors determine the relevance of a camera to the task of observing a pedestrian.

where $p(c_i|O)$ denotes the success probability of a camera c_i given task O.[2]

We describe the relevance of a camera to the task of observing a pedestrian in terms of the five factors listed in Fig. 1.3. Let $r(c_i, h_j)$ represent the relevance of a camera c_i to the task of recording closeup video of a pedestrian h_j, then

$$r(c_i, h_j) = \begin{cases} 1 & \text{if } c_i \text{ is idle;} \\ r_d r_\gamma r_{\alpha\beta\theta} r_o r_h & \text{otherwise,} \end{cases}$$

where

$$\begin{aligned} r_d &= \exp\left(-\frac{(d-\hat{d})^2}{2{\sigma_d}^2}\right), \\ r_\gamma &= \exp\left(-\frac{\gamma^2}{2{\sigma_\gamma}^2}\right), \\ r_{\alpha\beta\theta} &= \exp\left(-\frac{(\theta-\hat{\theta})^2}{2{\sigma_\theta}^2} - \frac{(\alpha-\hat{\alpha})^2}{2{\sigma_\alpha}^2} - \frac{(\beta-\hat{\beta})^2}{2{\sigma_\beta}^2}\right), \\ r_o &= \begin{cases} 1 & \text{if } \alpha \in [\alpha_{\min}, \alpha_{\max}] \\ & \quad \text{and } \beta \in [\beta_{\min}, \beta_{\max}] \\ & \quad \text{and } d < d_{\max}; \\ 0 & \text{otherwise,} \end{cases} \end{aligned}$$

$$r_h = \exp\left(-\frac{\varepsilon^2}{2{\sigma_\varepsilon}^2}\right).$$

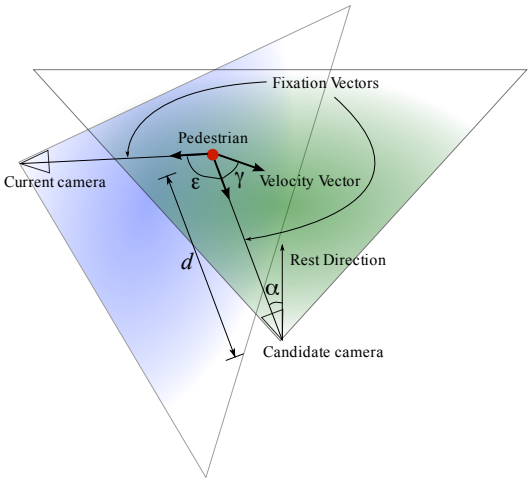


Figure 1.4: The relevance of a camera to the task of observing a person.

Here α and β are, respectively, the pan and tilt gaze angles corresponding to the 3D location of the pedestrian as computed by the triangulation process and θ corresponds to the field-of-view (zoom) setting required to capture closeup video of the pedestrian. Variables $\hat{\theta} = (\theta_{\min} + \theta_{\max})/2$, $\hat{\alpha} = (\alpha_{\min} + \alpha_{\max})/2$, and $\hat{\beta} = (\beta_{\min} + \beta_{\max})/2$, where $\theta_{\min}$ and $\theta_{\max}$ are extremal field-of-view settings, $\alpha_{\min}$ and $\alpha_{\max}$ are extremal vertical rotation pan angles, and $\beta_{\min}$ and $\beta_{\max}$ are extremal horizontal rotation tilt angles. Variable d denotes the camera-to-pedestrian distance, and $d_{\max}$ and $d_{\min}$ are the maximum and minimum distances at which the camera can reliably track a pedestrian. We set the optimal camera-to-pedestrian distance as $\hat{d} = (d_{\max} - d_{\min})/2$. The angle between the fixation vector of the camera and the velocity vector of the pedestrian is γ, and ε represents the angle between the fixation vector of camera c_i and the fixation vector of the camera currently observing the pedestrian (Fig. 1.4). The fixation vector (for a camera with respect to a pedestrian) is defined along the line joining the center of projection of the camera and the 3D position of the pedestrian. The values of the variances σ_d, σ_γ, σ_θ, σ_α, σ_β, and σ_ε associated with each attribute are chosen empirically; in our experiments, we set $\sigma_d = 10$, $\sigma_\gamma = \sigma_\theta = \sigma_\alpha = \sigma_\beta = 15.0$, and $\sigma_\varepsilon = 45.0$.

State Sequence Quality. The quality of a state sequence is

$$\mathcal{Q}(\mathcal{S}) = \prod_{t \in [0,1,\cdots]} \mathcal{Q}(s^t), \tag{1.1}$$

where the quality of a state s^t is determined by the success probabilities of individual PTZ cameras. Omitting superscript t for clarity,

$$\mathcal{Q}(s) = \prod_{i \in [1,N_a]} p(s_i) = \prod_{i \in [1,N_a]} r(c_i, h_j). \tag{1.2}$$

Rewriting (1.1), we obtain

$$\mathcal{Q}(\mathcal{S}) = \prod_{t \in [0,1,\cdots]} \left(\prod_{i \in [1,N_a]} r\left(c_i, h_j\right) \right). \tag{1.3}$$

Thus, $\mathcal{Q}(\mathcal{S})$ represents the probability of success of a state sequence $\mathcal{S}$ and it serves as a probabilistic objective function that enables the planner to compute state sequences (or plans) with the highest probability of success.

Planning

Finding an optimal state sequence is a combinatorial search problem, which typically cannot be carried out in real time. This is especially true for longer plans that arise in scenarios involving multiple pedestrians and larger networks of PTZ cameras. Camera control, however, must be carried out in real time. Therefore, planning activity must proceed in parallel with real-time camera control. In our case, the planning activity requires reliable predictions of the states (position and velocity) of pedestrians. Pedestrian state predictions are provided by the passive cameras. Obviously, the predictions become increasingly unreliable as the duration of the plan increases. It is therefore counterproductive to construct long plans.

We regard plans of length 10 or more as being long plans. The duration of a plan depends upon its length and the duration of each of its steps (in real-world time). We construct short plans consisting of action/state sequences of lengths between 5 and 10. When a new plan is available, actions are sent to the relevant PTZ cameras. Fig. 1.5 outlines our planning strategy.

Finding an Optimal Sequence

We employ *greedy best-first search* to find the optimal sequence of actions/states. The starting state along with the successor function, which enumerates all possible camera-pedestrian assignments, induce a state graph with branching factor $3^{h_s} N_a!/(N_a - h_s)!$, where h_s is the number of pedestrians selected to be observed by at least one PTZ camera. Fortunately, the branching factor is much smaller in practice due to the observational constraints of PTZ cameras and due to the preconditions imposed on camera actions (Table 1.1). Equation (1.2) provides the

Require: $\mathcal{A}_c$ {Current action sequence.}
Require: $\mathcal{S}_c$ {Current state sequence. Initially, $\mathcal{S}_c$ consists of a single state showing all PTZ cameras as idle.}
Require: t {Current time.}
Require: t_h {End time of the current plan.}
Require: t_b {Time budget available for planning.}
Require: t_p {Duration of the new plan.}
Require: Δt {Time step specifying the temporal granularity of the new plan.}
Require: e {Error flag indicating a re-planning request from a PTZ camera.}

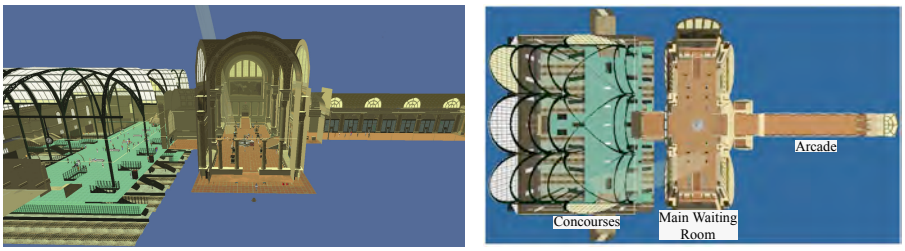
```
1: while Keep planning do
2:    while t + t_b < t_h and e = false do
3:       Update e {Check for any re-planning requests from PTZ cameras.}
4:    end while
5:    if e = false then
6:       s_0 ← EndState(S_c) {Last element of the current state sequence.}
7:    else
8:       s_0 ← CurrentState(S_c) {Re-planning starts from the current state.}
9:    end if
10:   (A*, S*) = Plan(s_0, t_b, Δt) {Find optimal action/state sequence starting from
      state s_0. Planning stops when the maximum plan depth t_p/Δt is reached or
      when the time budget for planning is exhausted.}
11:   if e = true then
12:      Replace(S_c, S*) {Replace current state sequence.}
13:      Replace(A_c, A*) {Replace current action sequence.}
14:   else
15:      Append(S_c, S*) {Append the new state sequence to the current state se-
         quence.}
16:      Append(A_c, A*) {Append the new action sequence to the current action se-
         quence.}
17:   end if
18:   e ← false {Reset error flag; essentially ignoring any errors that might have been
      raised by the PTZ cameras during the current planning cycle.}
19:   t_h ← t + t_p {Update end time (i.e., time horizon).}
20:   Send the new actions to the relevant PTZ cameras.
21: end while
```

Figure 1.5: The planning strategy for computing an optimal action/state sequence.

state value, whereas the path value is given by (1.3). To keep the search problem tractable, we compute short plans (comprising 5 to 10 steps). The time granularity Δt may be used to control the actual duration of a plan without affecting the associated search problem. For additional information on greedy best-first search, see [Russell and Norvig, 2003].

3. Results

Our visual sensor network is deployed and tested within our Virtual Vision train station simulator. The simulator incorporates a large-scale environmental model (of the original Pennsylvania Station in New



(a) (b)

Figure 1.6: (a) A cutaway side view of the virtual train station populated by autonomous, self-animating pedestrians. (b) Overhead view of the train station.

York City) with a sophisticated pedestrian animation system that combines behavioral, perceptual, and cognitive human simulation algorithms [Shao and Terzopoulos, 2007]. Standard computer graphics techniques enable a near-photorealistic rendering of the busy urban scene with considerable geometric and photometric detail (Fig. 1.6). Our companion chapter in this volume presents additional details about the simulator. In each of the following scenarios, passive wide-FOV cameras located in the virtual train station estimate the 3D positions of the pedestrians present in the scene.

Scenario 1: Fig. 1.7 shows a scenario consisting of 3 PTZ cameras that are tasked to record closeup video of a pedestrian as he makes his way through the shopping arcade towards the concourses in the train station. The camera network successfully accomplishes this goal. Initially, only Cam1 (shown as a blue triangle) is observing the pedestrian. Our planner anticipates that the pedestrian will soon enter the range of Cam2, and Cam2 is pre-tasked with observing the pedestrian, which results in a successful handoff between Cam1 and Cam2. As stated earlier, the planner constructs short-duration plans, so Cam3 is not considered at this time. During the next planning cycle, however, Cam3 is also taken into account as the pedestrian continues to walk towards the main waiting room. Cam3 and Cam2 perform a successful handoff.

Scenario 2: Fig. 1.8 depicts a far more challenging scenario, where 3 cameras are tasked to record closeup videos of two selected pedestrians. The first pedestrian (green trajectory) has entered the arcade and is moving towards the concourses, while the second pedestrian (cyan trajectory) has entered the main waiting room and, after purchasing a ticket at one of the ticket booths, is walking towards the arcade. Here, Cam3 temporarily takes over Pedestrian 1, thereby allowing Cam1 to handoff Pedestrian 2 to Cam2. Afterwards, Pedestrian 1 is handed off to Cam1.

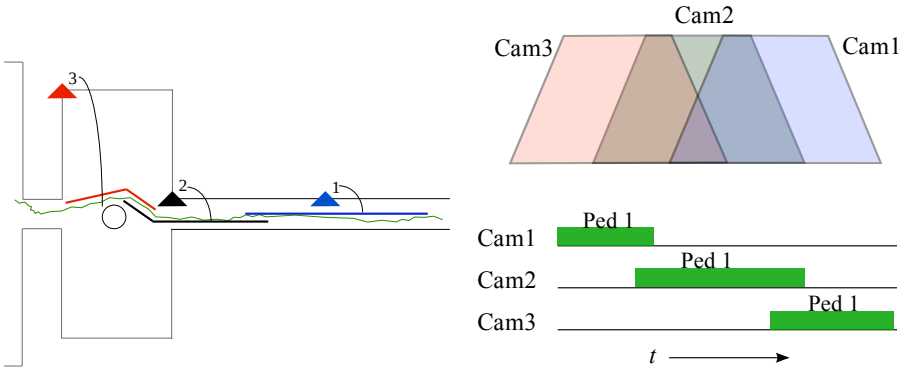


Figure 1.7: Scenario 1. Cameras 1, 2, and 3 perform handoffs to capture closeup video of the selected pedestrian. Outline (a) depicts the walls of the train station shown in Fig. 1.6.

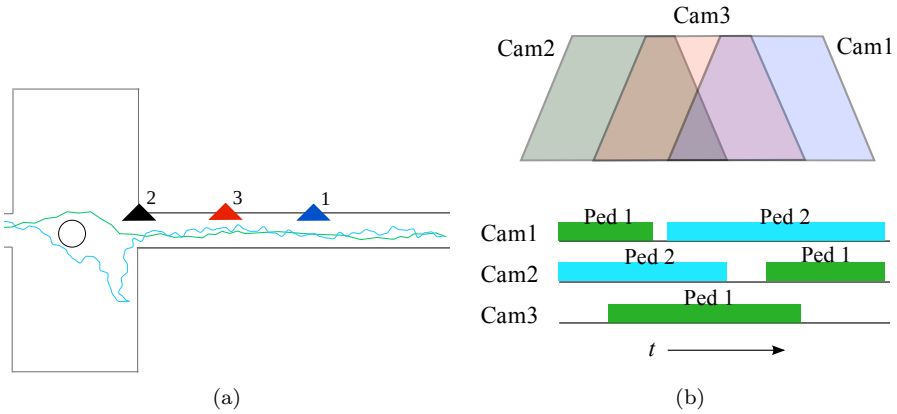


Figure 1.8: Scenario 2. Cameras 1, 2, and 3 successfully record closeup video of two selected pedestrians. Outline (a) depicts the walls of the train station shown in Fig. 1.6.

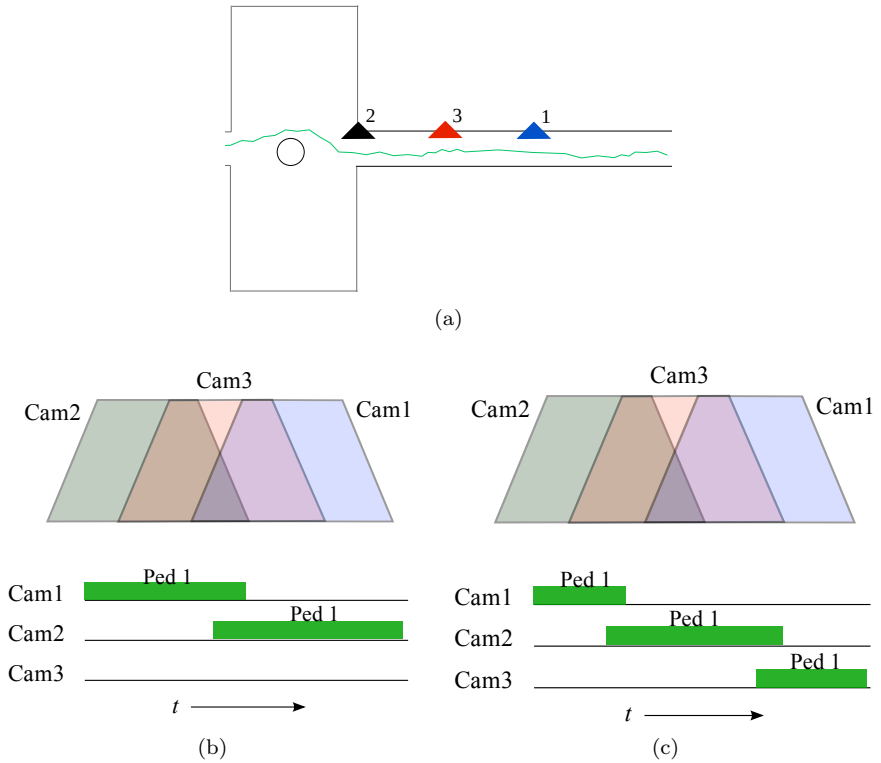
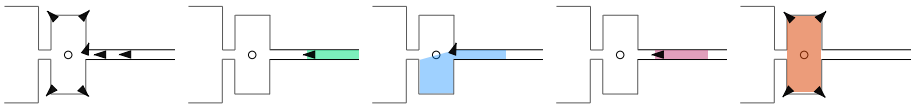


Figure 1.9: Scenario 3. Cameras 1, 2, and 3 successfully record closeup video of two selected pedestrians. Outline (a) depicts the walls of the train station shown in Fig. 1.6. The planner selects the strategy in (b), as it requires fewer handoffs.

Scenario 3: In Fig. 1.9, three PTZ cameras are tasked with recording closeup video of a pedestrian (green trajectory). Notice how the fields of view of all three cameras overlap; consequently, there is more than one correct handoff strategy, as shown in Fig. 1.9(b)–(c). The planner selects the handoff strategy in Fig. 1.9(b), as it requires fewer handoffs.

Scenario 4: Table 1.2 documents the success rates of capturing closeup videos of up to 4 pedestrians using a camera network comprising 7 PTZ cameras (shown in Fig. 1.10) plus passive wide-FOV cameras (not shown). A run is deemed successful if it satisfies the observation task—acquiring closeup videos of 1, 2, or 4 pedestrians while they remain in the designated area. The success rate is the ratio of the number of successful runs to the total number of runs. The results are aggregated over 5 runs each. As expected, when the network is tasked with closely observing a single pedestrian, the success rate is close to 100%; however, prediction errors prevent a flawless performance. When the network is



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

Figure 1.10: A virtual camera network deployed in our Virtual Vision simulator. (a) The positions of the virtual cameras. (b)–(d) The observational ranges of individual cameras. (e) The observational range of the four cameras situated at the corners of the main waiting room; cameras cannot observe/track any pedestrian outside their observational ranges.

# of Selected Pedestrians	**Short Plans** 5 Steps	**Long Plans** 10 Steps
1	99.8%	96%
2	95.1%	88%
4	67.2%	65.1%

Table 1.2: Success rates for the camera network shown in Fig. 1.10.

tasked with simultaneously observing 2 pedestrians, the success rate falls to 95.1% for short-duration plans and it is below 90% for long-duration plans. Again, we can attribute this behavior to errors in predicting the state of the selected pedestrians. Next, the camera network is tasked to observe 4 pedestrians simultaneously. The success rate now falls to 67% for short-duration plans and 65% for long-duration plans. This is partly due to the fact that the planner cannot find an admissible state sequence when the four pedestrians aggregate in the arcade.

4. Conclusions and Future Work

We have described a planning strategy for intelligently managing a network of active PTZ cameras so as to satisfy the challenging task of capturing, without human assistance, closeup biometric videos of selected pedestrians during their prolonged presence in an extensive environment under surveillance. The ability to plan ahead enables our surveillance system to avoid camera assignments that might appear optimal at present, but will later lead to observation failures. The planning process assumes the reliable prediction of pedestrian states, which is currently provided by the supporting stationary wide-FOV passive cameras. We have noticed that short duration plans are preferable to longer duration plans as 1) state predictions are less reliable for longer plans, and 2) longer plans take substantially longer to compute, which adversely affects the relevance of a plan when it is executed.

Scalability is an issue when dealing with numerous active cameras spread over an extensive region. In the long run, we hope to tackle the scalability issue by investigating distributed multiagent planning strategies. In the shorter term, we will address the scalability issue by restricting planning activity to the relevant cameras by first grouping cameras with respect to the active tasks. Our strategy assumes a fixed camera setup; it currently does not support *ad hoc* camera deployment, a limitation that we intend to address in the future. We have prototyped our surveillance system in a virtual train station environment populated by autonomous, lifelike pedestrians. However, we intend to evaluate our planning strategy using a physical camera network, which will involve additional technical challenges.

Acknowledgments

The work reported herein was supported in part by a UOIT Startup Grant and an NSERC Discovery Grant. We thank Wei Shao and Mauricio Plaza-Villegas for their invaluable contributions to the implementation of the Penn Station simulator.

Notes

1. A good compromise is to restrict the planning to the group of "relevant" cameras.
2. Ideally, $p(c_i|O) = \mathcal{F}\left(r(c_i, O)\right)$, where function $\mathcal{F}$ should be learned over multiple trials. Krahnstoever *et al.* arrive at a similar conclusion [Krahnstoever et al., 2008].

References

Collins, R., Amidi, O., and Kanade, T. (2002). An active camera system for acquiring multi-view video. In *Proc. International Conference on Image Processing*, pages 517–520, Rochester, NY.

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

Farrell, R. and Davis, L. S. (2008). Decentralized discovery of camera network topology. In *Proc. Second International Conference on Distributed Smart Cameras (ICDSC08)*, Menlo Park, CA.

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

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

Heath, K. and Guibas, L. (2008). Multi-person tracking from sparse 3D trajectories in a camera sensor network. In *Proc. Second International*

Conference on Distributed Smart Cameras (ICDSC08), Menlo Park, CA.

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

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

Krahnstoever, N. O., Yu, T., Lim, S N, Patwardhan, K., and Tu, P. H. (2008). 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.

Li, Y. and Bhanu, B. (2008). Utility-based dynamic camera assignment and hand-off in a video network. In *Proc. Second International Conference on Distributed Smart Cameras (ICDSC08)*, pages 1–9, Menlo Park, CA.

Meijer, P. B. L., Leistner, C., and Martiniere, A. (2007). Multiple view camera calibration for localization. In *Proc. First IEEE/ACM International Conference on Distributed Smart Cameras (ICDSC07)*, pages 228–234, Vienna, Austria.

Park, J., Bhat, P. C., and Kak, A. C. (2006). A look-up table based approach for solving the camera selection problem in large camera networks. In Rinner, B and Wolf, W, editors, *Working Notes of the International Workshop on Distributed Smart Cameras (DSC 2006)*, pages 72–76, Boulder, CO.

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

Qureshi, F. Z. and Terzopoulos, D. (2006). Surveillance camera scheduling: A virtual vision approach. *ACM Multimedia Systems Journal*, 12(3):269–283.

Qureshi, F. Z. and Terzopoulos, D. (2008). Smart camera networks in virtual reality. *Proceedings of the IEEE (Special Issue on Smart Cameras)*, 96(10):1640–1656.

Russell, S. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Pearson, Prentice Hall Series in Artificial Intelligence, 2nd edition.

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

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

Index