Opportunistic Sensing in a Distributed PTZ Camera Network

Chong Ding, Akshay A. Morye, Jay A. Farrell, Amit K. Roy-Chowdhury
Bourns College of Engineering
University of California, Riversde

***Abstract*—The performance of video-based scene analysis algorithms often suffers because of the inability to effectively acquire features on the targets. In this paper, we propose a distributed approach for dynamically controlling the pan, tilt, zoom (PTZ) parameters of a PTZ camera network so as to maximize system performance, through opportunistic acquisition of high quality images. The cameras gain utility by achieving the tracking specification and through high resolution feature acquisition. High resolution imagery comes at a higher risk of losing the target in a dynamic environment due to the corresponding decrease in the field of view (FOV). This optimization will determine not only *how* the cameras are controlled, but also *when* to obtain high quality images. The target state estimates, upon which the control algorithm is dependent, are obtained through a distributed tracking algorithm. Our approach is developed within a Bayesian framework to appropriately trade-off value (target tracking accuracy and image quality) versus risk (probability of losing track of a target). This article presents the theoretical solution along with simulation and experimental results on a real camera network.**

I. Introduction

Camera networks are rich information sources for tasks related to security and surveillance, environmental monitoring, disaster response, etc. Many applications require distributed processing over the network as traditional centralized processing scales poorly due to network bandwidth constraints. Distributed vision systems can enhance or enable operations in places without pre-existing network infrastructure, like search and rescue operations and environmental monitoring. This leads to a multi-agent network of cameras, where the individual cameras need to actively coordinate between themselves to sense, learn and reason about the environment. Many recent papers such as [1], [2] and [3] have been dedicated to developing distributed versions of computer vision algorithms.

These agents are given some high-level objectives and rules to perform certain tasks. As an example, the network might be tasked with tracking all moving objects, obtaining identity information for each, and understanding their behaviors. The rules entail certain video analysis tasks that need to be performed; for example, tracking involves obtaining the positions of the targets in the 3D world, person recognition is obtained from frontal facial shots, understanding behaviors means obtaining high resolution shots of the entire person or groups of people when they are in close proximity. The goal of this paper is to combine these video analysis problems with multi-agent control mechanisms to provide high quality imagery for analysis.

A. Related Work

The research presented here is related to a classical problem of computer vision, namely active vision [4]. However, active vision in a distributed camera network, where the cameras coordinate among themselves, is still relatively unexplored. Many modern vision networks consist of a mixture of static and PTZ cameras. The placement of these cameras is determined at the moment of deployment. Optimal camera placement strategies were proposed in [5] and solved by using a camera placement metric that captures occlusion in 3-D environments, and binary integer programming. In [6], a solution to the problem of optimal camera placement given some coverage constraints was presented and can be used to come up with an initial camera configuration.

In many mixed camera networks, the static cameras and PTZ cameras are assigned different tasks. The path planning inspired approach proposed by [7] used static cameras to track all targets in a virtual environment while PTZ cameras were assigned to obtain high resolution video from the targets. This approach showed that given the predicted tracks of all the targets, a set of one-to-one mappings between cameras and targets can be formed to acquire high resolution videos. A method for determining good sensor configurations that would maximize performance measures was introduced in [8]. The configuration framework was based on the presence of random occluding objects and two techniques were proposed to analyze the visibility of the objects. These methods address the camera network reconfiguration problem in a centralized manner and may not be ideal for applications constrained by bandwidth and power.

A recent distributable approach in [9] uses the Expectation-Maximization (EM) algorithm to find the optimal configuration of PTZ cameras given a map of activities. The value of each discretized ground coordinate is determined using the map of activities. This approach upon convergence of the EM algorithm, provides the PTZ settings to optimally cover an area given the map of activities. The authors in [10], [11] showed how cameras can coordinate between themselves to perform area coverage. Preliminary work in target tracking with the ability to obtain high resolution shots was shown in [12] for a single timestep simulation.

The distributed optimization approach proposed here can be mapped to the EM approach used in [9]. The prediction of the global utility is comparable to the expectation step and

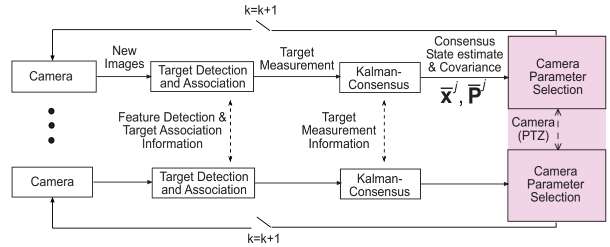

Fig. 1. Diagram depicting the framework for integrating scene analysis and PTZ control.

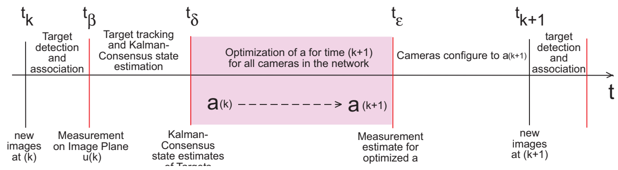

Fig. 2. Timeline of events between image sample times.

the maximization of the camera utility can be related to the maximization step in the EM approach. In these methods the cameras coordinate between themselves to always meet the objective. However, as the complexity of the system increases and there are multiple objectives to be satisfied, it is highly unlikely that each objective can be met at every point of time. Nor is it necessary as the design of such a system would be a huge waste of resources. Our method differs by *opportunistically* aquiring images that satisfy each objective. For example, since a single passport quality image of the face is usually enough for recognition, we design our optimization function so that we gain utility only when a higher quality image is able to be obtained.

II. Solution Overview

Our solution shown in Fig. 1 can find application in any visual surveillance system that contains PTZ cameras. PTZ configurations with large FOVs can monitor a large area of the enviroment but may not supply reliable images for recognition tasks. On the other hand, PTZ configurations pointed and zoomed in on specific areas of interest can gather useful images for recognition, but have a much smaller view of the scene. The proposed work is based on the idea that PTZ cameras can be automatically scheduled to satisfy multiple objectives based on the state of the scene and the state of the video surveillance system. Thus, the video streams are processed in order to detect and track moving objects; these are then processed to determine the best collection of PTZ parameters for the following time instant. A timeline of these events can be seen in Fig. 2. We assume each PTZ camera in our network is capable of low level video processing. In a large network, the ability to process video data locally significantly reduces bandwidth consumption. A distributed solution where each camera is responsible for incorporating its local data would eliminate the problems associated with transmitting many simultaneous video feeds to a central server. We maintain this view as we build our functions for evaluating the PTZ settings available to each camera. We show results for a surveillance system comprised of PTZ cameras tasked with tracking and recognition of people over a wide area.

III. Methodology

Our task has many similarities to problems in cooperative game theory, in which players' decisions are based on a utility function and the actions of the other players: in our case each camera decides on a configuration given its local information and the choices of the other cameras. One of the more popular methods to solve such cooperative games is Nash equilibria [13].

The design cooperative games parallels the theme of distributed optimization and can be thought of as a concatenation between a designed game and a distributed learning algorithm [14]. We give a short description of cooperative games followed by how it can be applied to active sensing in a distributed camera network.

A. Cooperative Games and Camera Reconfiguration

The concept of Nash equilibrium is where each player makes the best decision, taking into account the best decisions of every other player. While there are many games where this analysis may be applied, we describe the special case called the *ordinal potential game*. In such a game, the incentive of all players to change their actions can be expressed in a global potential function Φ [13]. Also, any positive change in the local utility results in a positive change in the global potential function. This allows us to maximize the global potential function by maximizing the local utility of each player in turn.

This cooperative control approach has recieved significant attention in recent years, mostly in the design of autonomous vehicles; an autonomous vehicle-target assignment problem was solved using this approach in [15]. The question is how can this analysis be applied to active sensing in a camera network? The basic idea is that the potential function Φ represents the performance of the system as a whole, in relation to the goals of tracking, identifying and interpreting the interactions of people in the area under surveillance. By viewing each camera as a player, its local utility U_{C_i} can be defined as equivalent to its contribution to the global system performance Φ. More formally, if $\mathbf{a} \in S$, where S is the collection of all possible camera PTZ settings in the game G, and S_i is the collection of all possible camera PTZ settings for camera C_i, then the function $\Phi(\mathbf{a}) : S \rightarrow \Re$ is an ordinal potential function for the game G, if $\forall \mathbf{a} \in S$ and $\forall \mathbf{a}_i, \mathbf{b}_i \in S_i$,

$$
\begin{aligned}
&U_{C_i}(\mathbf{b}_i, \mathbf{a}_{-i}) - U_{C_i}(\mathbf{a}_i, \mathbf{a}_{-i}) > 0 \\
&\Rightarrow \Phi(\mathbf{b}_i, \mathbf{a}_{-i}) - \Phi(\mathbf{a}_i, \mathbf{a}_{-i}) > 0 \qquad (1)
\end{aligned}
$$

where, $\mathbf{a}_{-i}$ is the set of camera PTZ settings excluding the settings for camera C_i and $\Delta U_{C_i}(\mathbf{a}) > 0$ makes the game a maximum game. This allows us to maximize the system performance through the maximization of the local utility of each camera according to Algorithm 1.

Algorithm 1 Distributed Optimization Strategy

Input: Camera C_i
C_i calculates parameters needed to maximize $U_{C_i}(\mathbf{a}_i)$ based on the proposed settings of the other cameras
if C_i needs to change its parameters to maximize its utility **then**
  C_i changes its parameters
  C_i broadcasts its parameters to all cameras in the network
end if

B. Potential and Utility Functions

Now that we have mapped our problem into the domain of potential games, we need to design a resonable $\Phi(\mathbf{a})$ and corresponding $U_{C_i}(\mathbf{a})$ given the goals of our system. The basis for many identification or recognition algorithms is a series of high resolution images of a particular feature or set of features. Aquiring such images requires the targets to be well tracked, so that there is little possibility of failing to aquire a target or feature when zooming in. We begin by describing the measurement model, followed by the design of utility functions representative of each goal. Finally, we will show how the global and local utilities are designed.

1) Measurement Model: This section descibes the measurement model for target j by camera i. We assume that an estimate ${}^{w}\hat{\mathbf{p}}^j$ of the target position in the world is available, that ${}^{w}\mathbf{p}_{c_i}$ is known, and that the rotation from the world to the i-th camera frame ${}^{c_i}_{w}\mathbf{R}$ is known. The rotation matrix ${}^{c_i}_{w}\mathbf{R}(\rho_i, \tau_i)$ is dependent on the pan angle ρ_i and tilt angle τ_i selected for camera i and the focal length F_{c_i} is dependent on the zoom setting ζ_i. We use the notation ${}^{c_i}\mathbf{p}^j = \left[{}^{c_i}x^j, {}^{c_i}y^j, {}^{c_i}z^j\right]^\top$ for the position of the j-th target in the camera frame.

The j-th target's position in the camera frame is related to the position of that target in the global frame by

$$\begin{bmatrix} {}^{c_i}\mathbf{p}^j \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{c_i}_{w}\mathbf{R} & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} & -{}^{w}\mathbf{p}_{c_i} \\ \mathbf{0}^\top & 1 \end{bmatrix} \begin{bmatrix} {}^{w}\mathbf{p}^j \\ 1 \end{bmatrix}, \quad (2)$$

where $\mathbf{0} \in \Re^{3\times 1}$ and $\mathbf{I}$ is the identity matrix in $\Re^3$.

In homogenous coordinates, the i-th camera's image plane coordinates for the j-th target, ${}^{im}\mathbf{p}^j = [{}^{im}x_h, {}^{im}y_h, {}^{im}w]^\top$, are

$${}^{im}\mathbf{p}^j = \begin{bmatrix} -\frac{F_{c_i}}{s_x} & 0 & 0 & o_x \\ 0 & -\frac{F_{c_i}}{s_y} & 0 & o_y \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} {}^{c_i}\mathbf{p}^j \\ 1 \end{bmatrix}. \quad (3)$$

In Eqn. (3), ${}^{im}\mathbf{p}^j$ are the homogenous coordinates of the target's position on the image plane with weight ${}^{im}w$. The symbols o_x and o_y represent the coordinates of the location of the image center in pixel coordinates, s_x and s_y represent the effective pixel size in the horizontal and vertical direction, respectively, and F_{c_i} is the focal length of the i-th camera.

The pixel coordinates of the target's position can then be determined by

$${}^{im}\mathbf{u}^j = f({}^{im}\mathbf{p}^j) = \begin{bmatrix} \frac{{}^{im}x_h}{{}^{im}w} \\ \frac{{}^{im}y_h}{{}^{im}w} \end{bmatrix}. \quad (4)$$

Accounting for noise, the measurement from the i-th camera is

$${}^{im}\tilde{\mathbf{u}}^j = {}^{im}\mathbf{u}^j + {}^{im}\boldsymbol{\eta} \quad (5)$$

where we assume that ${}^{im}\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}_{2\times 1}, \mathbf{Q}_i)$.

2) Tracking Utility: The purpose of the tracking utility is to quantify how well camera C_i believes the system will track target T^j given the proposed settings for all cameras. As we are interested in a fully distributed system, we assume that a distributed tracker, capable of providing a fused state ${}^{w}\bar{\mathbf{x}}^j$ and error covariance $\mathbf{P}^j$ for each target, is present. For our experiments we used the Kalman-consensus tracker from [16].

The state of each target is represented in the world frame and contains both the position and velocity and is represented as ${}^{w}\bar{\mathbf{x}}^j = [{}^{w}\mathbf{p}^j, {}^{w}\mathbf{v}^j]$, where ${}^{w}\mathbf{p}^j = [x, y]$ and ${}^{w}\mathbf{v}^j = [v_x, v_y]$. The error covariance matrix $\mathbf{P}^j$ can be represented in block form as

$$\mathbf{P}^j = \begin{bmatrix} \mathbf{P}^j_{\mathbf{pp}} & \mathbf{P}^j_{\mathbf{pv}} \\ \mathbf{P}^j_{\mathbf{vp}} & \mathbf{P}^j_{\mathbf{vv}} \end{bmatrix}, \quad (6)$$

where $\mathbf{P}^j_{\mathbf{pp}}$ represents the position error covariance matrix. Assuming calibration we can compute the linearized transformation matrix $\mathbf{H}^j_i$ [17] between the world and image for each camera. As $\mathbf{H}^j_i$ is a function of the PTZ of the camera and target we can evaluate the expected error covariance of the target given the proposed PTZ settings of all cameras as

$$\mathbf{P}^{j+} = \left((\mathbf{P}^{j-})^{-1} + \sum_{i=1}^{N_C} \mathbf{H}^{j\top}_i (\mathbf{Q}^j_i)^{-1} \mathbf{H}^j_i \right)^{-1}, \quad (7)$$

where N_C is the number of cameras viewing target T^j. The corresponding posterior information matrix is denoted as $\mathbf{J}^{j+} = \left(\mathbf{P}^{j+}\right)^{-1}$. We now define the tracking utility as the average trace of the information of all targets,

$$U_T(\mathbf{a}) = \frac{1}{N_T} \sum_{j=1}^{N_T} trace(\mathbf{J}^{j+}) \quad (8)$$

3) Imaging Utility: The purpose of the imaging utility is to determine whether the resolution and/or pose requirements of target T^j or features of target T^j are satisfied by the camera network using settings profile $\mathbf{a}$. Thus, the imaging utility is comprised of view angle and resolution coefficients. In many instances, only a few high quality images per target are sufficient for recognition . Once one such image is acquired for T^j, then the utility contributed by the imaging utility should have added value only if a better image can be procured at a new viewing angle $\theta^{j_n}_i$ that is closer to the optimal view angle $\bar{\theta}^j$, or at a higher resolution, or both.

Imaging at Specified Pose: Let T^j maneuver in the area with a direction vector $\mathbf{o}_{T^j}$. Defining a vector $\mathbf{o}_{C_i}$ from camera C_i's position ${}^{w}\mathbf{p}_i$ to T^j's estimated position ${}^{w}\hat{\mathbf{p}}^j$, we can compute the view angle θ^j_i formed by T^j at C_i as,

$$\theta^j_i = \arccos\left(\frac{\mathbf{o}_{T^j} \cdot \mathbf{o}_{C_i}}{\|\mathbf{o}_{T^j}\| \|\mathbf{o}_{C_i}\|} \right) \quad (9)$$

where all viewing angles are between 0 and 2π. An illustration of the view angle factor is shown in Fig. 3. Let us assume that

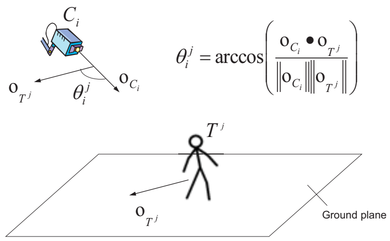

Fig. 3. Camera viewing angle and Target pose.

at a previous time, a high resolution image of T^j was obtained at viewing angle θ^{j_p}. Let $\theta_i^{j_n}$ be the angle at which camera C_i can procure a new image of T^j. Defining a view angle coefficient m_i^θ that weighs $U_I^j(\mathbf{a})$ to provide a higher value for a new viewing angle $\theta_i^{j_n}$ closer to the desired angle $\bar{\theta}^j$ than the previous viewing angle θ^{j_p},

$$m_i^\theta = \begin{cases} \frac{\left|\theta^{j_p} - \left|\bar{\theta}^j - \theta_i^{j_n}\right|\right|}{\left|\bar{\theta}^j - \theta^{j_p}\right|} & \text{if } \left|\theta^{j_p} - \left|\bar{\theta}^j - \theta_i^{j_n}\right|\right| > \left|\bar{\theta}^j - \theta^{j_p}\right| \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

Assuming that the target is facing in the direction of motion and $\bar{\theta}^j = 0$, Eqn.(10) provides a positive non-zero value of m_i^θ, only when target T^j is moving towards the camera. Thus, the scalar m_i^θ measures alignment of the camera-to-target vector with the target's direction vector.

Imaging at Specified Resolution: We then define a resolution coefficient m_i^r as a measure of viewing target T^j at desired resolution $\bar{r}$.

$$m_i^r = \begin{cases} \frac{r_i^j - \underline{r}}{\bar{r} - \underline{r}} & \text{if } \underline{r} < r_i^j > \bar{r} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

where $\underline{r}$ and $\bar{r}$ are the minimum and maximum height requirements of target T^j in the camera image plane, and r_i^j is the resolution at which T^j or feature of T^j is being viewed at by C_i.

Thus, we can now define an imaging utility $U_I^j(\mathbf{a})$ for a given pose and resolution requirement as:

$$U_I^j(\mathbf{a}) = \max_i \left(m_i^\theta m_i^r\right) \tag{12}$$

The maximization of $U_I^j(\mathbf{a})$ across all targets T^j would thus lead to a set of PTZ settings resulting in the best set of images given the pose and resolution requirements of each target.

4) Global Potential Function: We can now define the global potential $\Phi(\mathbf{a})$ representing the system goals as:

$$\Phi(\mathbf{a}) = U_T(\mathbf{a}) + \sum_{j=1}^{N_T} g\left(U_T\right) U_I^j w^j(\mathbf{a}). \tag{13}$$

In this definition, $U_I^j(\mathbf{a})$ is a function that rewards high resolution face images of T^j. The function g is a continuously differentiable monotonically increasing bounded function and w^j is a possibly time varying weight that magnifies the importance of imagery for certain targets relative to others. The function $g\left(U_T\right)$ is defined as:

$$g\left(U_T\right) = \frac{1}{1 + \exp\left(\kappa_g\left(\bar{P} - U_T\right)\right)}, \tag{14}$$

where $\bar{P}$ is the tracking accuracy threshold. Such a choice of g, for large κ_g, ensures that the maximization of $U_I^j(\mathbf{a})$ for any target is only factored in under the condition that all coordinates of all targets are expected to exceed the accuracy specified by $\bar{P}$. If this condition is not satisfied, the second term in Eqn.(13) is near zero. High priority is given to obtaining high quality facial images, once all targets are tracked to an accuracy better than $\bar{P}$.

Assuming quality of image capture to be a function of the number of pixels on the target being imaged, it is desirable to have $U_I^j(\mathbf{a})$ as a monotonically increasing function but only until an imaging threshold $\bar{r}(\mathbf{a})$ is met. Let the threshold $\bar{r}(\mathbf{a})$ be a function of the maximum number of pixels permissible on the target in the image for efficient target recognition. Subsequently, $U_I^j(\mathbf{a})$ should monotonically decrease. Various choices are possible for $U_I^j(\mathbf{a})$ depending on the desired behavior, one of which has been defined in Eqn.(12).

5) Bayesian Value $V(\mathbf{a})$: Because the global utility $\Phi(\mathbf{a})$ that is actually received is dependent on the random variables ${}^w\mathbf{p}^j(k+1)$ for $j = 1, \ldots, N_T$, through $\mathbf{H}_i^j$ and the FOV, the global utility is a random variable. Therefore, the optimization will be based on the expected value of the global potential function $\Phi(\mathbf{a})$ over the distribution of the uncertainty in the estimated position of the target.

Hence, we define a Bayesian value function $V(\mathbf{a})$ as:

$$\begin{aligned} V(\mathbf{a}) &= E\left\langle \Phi(\mathbf{a}; {}^w\mathbf{p}^j,\ j = 1, \ldots, N_T)\right\rangle && (15) \\ &= \int \Phi(\mathbf{a}) p_{\mathbf{p}}\left(\boldsymbol{\zeta}\right) d\boldsymbol{\zeta} && (16) \end{aligned}$$

The dummy variable $\boldsymbol{\zeta}$ is used for integration over the ground plane and $p_{\mathbf{p}}$ is the Normal distribution $\mathcal{N}({}^w\hat{\mathbf{p}}^j, \mathbf{P}_{\mathbf{pp}}^{j-})$ of the predicted position of T^j in the global frame at the next imaging instant, where $\mathbf{P}_{\mathbf{pp}}^{j-}$ represents the position covariance matrix.

The integral represents the area spanned by the FOV of the camera. Inside the integral, the global utility $\Phi(\mathbf{a})$ is multiplied by the probability distribution function $p_{\mathbf{p}}$, where the maximum value for $p_{\mathbf{p}}$ occurs at the estimated target position ${}^w\hat{\mathbf{p}}^j$. Thus, integrating over the FOV makes the camera C_i select a settings profile $\mathbf{a}_i$ such that most of the ellipsoid formed by the position covariance matrix $\mathbf{P}_{\mathbf{pp}}^{j-}$ around the position estimate ${}^w\hat{\mathbf{p}}^j$, is in view, thus reducing the risk of not imaging the target.

6) Local Camera Utility Function: All that remains is to appropriately design the local camera utilty such that it satisfies Equation 1. This can be achieved by decoupling $V(\mathbf{a})$ into the contributions made by camera C_i and all the other C_{-i} cameras and can be written as,

$$U_{C_i}(\mathbf{a}_i) = V(\mathbf{a}) - V(\mathbf{a}_{-i}). \tag{17}$$

We can then maximize the global potential function by solving for Nash equilibrium according to Algorithm 1.

C. Applications

The utility functions defined above can be used in a number of different applications. For face recognition we set the $\bar{\theta}^j = 0$ as facial images are best obtained from the front. For interactions such as handing over objects, shaking hands and waving, a high resolution side view of the targets provide more discernable images. This can be easily handled by our existing imaging utility defined in Section. III-B3 by setting the pose requirement $\bar{\theta}^j = \pi$ and the target resolution requirements $\underline{r}$ and $\bar{r}$. In cases where only resolution of the feature matters, we could set $m_i^\theta = 1$. This would make the imaging utility a function of only resolution.

While the global potential function defined above is for two goals, it can also be easily modified to add an additional system goals. For example, if we wanted to recognize interactions between targets in addition to tracking and identification, we could add another imaging utility to $\Phi(\mathbf{a})$.

IV. Experimental Results

Conducting experiments on real distributed camera networks can be very challenging; controlling these cameras autonomously further complicates things as every component needs to work, and in real-time. We used roof mounted AXIS PTZ 215 cameras connected though a Wireless-G network. As we did not have the developmental resources to develop a fully distributed embedded system, the video data was streamed to a PC where each camera operated within its own software thread. Thus, the experimental framework is still limited by the bandwidth and processing constraints of traditional systems.The performance is also reliant on the performance and capability of the algorithms chosen for detecton and tracking. In this experiment we used a simple background subtraction based detector in conjunction with the distributed Kalman-consensus tracker mentioned earlier. Fig.(4) shows the typical images aquired when optimizing the tracking performance only (i.e, imaging utility is not considered). We can see that as tracked targets move through the area higher resolution images may be aquired to improve the tracking accuracy. However these high resolution full body images may have insufficient resolution and pose for identification based on facial images.

We now show how cameras in a distributed network collaborate to decide *how* and *when* to obtain high quality images of features. The orientation of the target is determined by its estimated velocity vector if the magnitude is above 10 *cm/s*. Otherwise it is assumed to be unknown and the $U_I^j(\mathbf{a})$ for that target will be 0. The face was assumed to be located in the topmost 40 *cm* of the target's height and the expected resolution is the height of this region in pixels. We setup a sensor network of three calibrated PTZ cameras and $N_T = 4$ targets, located in and around the area. As H_i^j is dependent on the camera parameters, we calibrated each camera at a particular setting and the rest of the homography matrices were determined by modifying the values in Eqn. (2) and (3). The

(a)

(b)

Fig. 4. The images in (a) and (b) are captured by our real life system with $N_C = 4$ cameras. These images are the typical results we get when the tracking performance for targets is being optimized. What is important here is that while higher resolution images of different targets may be acquired to improve tracking, the faces are very hard to make out and are not very good for identification purposes.

results are shown for a period of $T = 50$ seconds. All cameras were set to resolution of 320×240 pixels.

The plots of the utility functions along with the tracking error covariance and the resolution of the faces are shown in Fig.(5). By using only $N_C = 3$ cameras to maintain coverage, many situations where the tracking threshold $\bar{P}$ is not met can be clearly seen in the $g(U_T)$ plot in Fig.(5). At time-step t_{11} the pose and resolution requirement for a target is satisfied and thus a high value for $U_I(\mathbf{a})$ is seen. This results in capture of a high quality face image at time-step t_{12} of T^1 at $48.0°$ from the desired angle. It also leads to reduction in tracking performance due to having one less camera generating measurements, which can be seen as a reduction in $U_T(\mathbf{a})$. Another high-resolution image is obtained at t_{28} for target T^1,

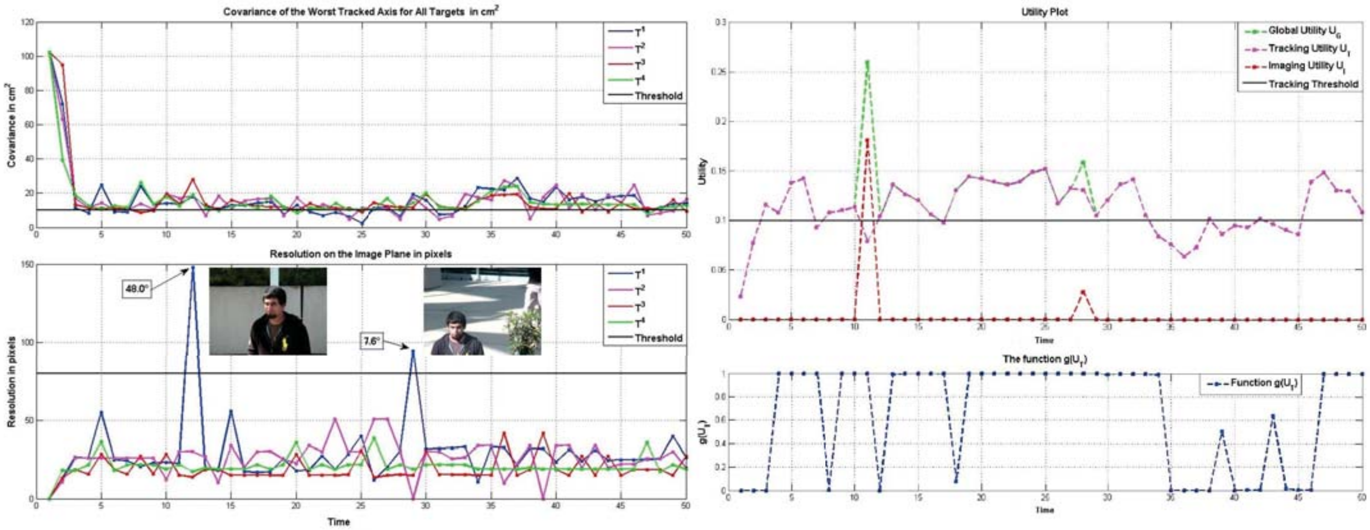

Fig. 5. Plots of utilities, tracking covariance, image resolution, and the function $g(U_T)$, for $N_C = 3$ cameras, to track and image $N_T = 4$ targets. The global potential $\Phi(\mathbf{a})$ represents the summation of tracking and imaging utilities. The tracking utility $U_T(\mathbf{a})$ is a measure of the tracking performance of the least accurately tracked target. When $U_T(\mathbf{a})$ satisfies tracking threshold $\bar{P}$, a non-zero value for the function $g(U_T)$ is obtained. If pose and resolution requirements for imaging the target are satisfied, then a spike for the imaging utility $U_I(\mathbf{a})$ can be seen.

at angular distance of $7.6°$. The second high-resolution image for the same target is obtained, due to an improvement over the previous viewing angle. This leads also to degradation in tracking. But, in spite of degradation in $U_T(\mathbf{a})$, it stays above $\bar{P}$, thus enabling $g(U_T)$ to have a non-zero value. We can see that the face images acquired are of much better quality than those shown earlier in Fig.(4).

V. Conclusion

In this article, we proposed a method to prioritize tasks for a distributed camera network to co-operatively track all targets and procure high resolution images, when the opportunity arises, subject to target pose and other criteria. We designed utility functions to evaluate the PTZ settings of the cameras for both tracking and feature imaging. These were used within our Bayesian distributed optimization framework to select optimal PTZ settings for the camera network at every time instant. We also showed in our results how our designed utilities reduced the resources required to enable high quality feature acquisition.

Acknowledgment

This work was partially supported by NSF grant 0622176 and ONR grant N00014-09-1-0666.

References

[1] R. Tron and R. Vidal, "Distributed Algorithms for Camera Sensor Networks," *IEEE Signal Processing Magazine*, vol. 3, pp. 32–45, May 2011.

[2] M. Taj and A. Cavallaro, "Distributed and Decentralized Multicamera Tracking," *IEEE Signal Processing Magazine*, vol. 3, pp. 46–58, May 2011.

[3] B. Song, C. Ding, A. Kamal, J. Farrell, and A. Roy-Chowdhury, "Distributed wide area scene analysis in reconfigurable camera networks," *IEEE Signal Processing Magazine*, vol. 3, pp. 20–31, May 2011.

[4] J. Aloimonos, I. Weiss, and A. Bandopadhay, "Active vision," *International Journal of Computer Vision*, no. 2, 1988.

[5] J. Zhao, S. C. Cheung, and T. Nguyen, "Optimal Camera Network Configurations for Visual Tagging," *IEEE Journal on Selected Topics in Signal Processing Special Issue on Distributed Processing in Vision Networks*, August 2008.

[6] U. M. Erdem and S. Sclaroff, "Automated camera layout to satisfy task-specific and floor plan-specific coverage requirements," *Comput. Vis. Image Underst.*, vol. 103, no. 3, pp. 156–169, 2006.

[7] F. Qureshi and D. Terzopoulos, "Planning Ahead for PTZ Camera Assignment and Handoff," in *IEEE/ACM Intl. Conf. on Distributed Smart Cameras*, Como, Italy, Aug-Sep 2009, pp. 1–8.

[8] A. Mittal and L. Davis, "A general method for sensor planning in multi-sensor systems: Extension to random occlusion," *International Journal of Computer Vision*, vol. 76, pp. 31–52, 2008.

[9] C. Piciarelli, C. Micheloni, and G. Foresti, "PTZ camera network reconfiguration," in *IEEE/ACM Intl. Conf. on Distributed Smart Cameras*, Como, Italy, Aug. 2009, pp. 1–8.

[10] C. Soto, B. Song, and A. Roy-Chowdhury, "Distributed Multi-Target Tracking In A Self-Configuring Camera Network," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[11] C. Ding, B. Song, A. Morye, J. Farrell, and A. Roy-Chowdhury, "Collaborative sensing in a distributed ptz camera network," *Image Processing, IEEE Transactions on*, vol. 21, no. 7, pp. 3282 –3295, july 2012.

[12] A. Morye, C. Ding, B. Song, A. Roy-Chowdhury, and J. Farrell, "Optimized Imaging and Target Tracking within a Distributed Camera Network," in *American Control Conference*, 2011.

[13] D. Monderer and L. Shapley, "Potential games," *Games and Economic Behavior*, no. 1, May 1996.

[14] N. Li and J. Marden, "Designing games for distributed optimization," in *IEEE Conf. on Decision and Control*, Florida, USA, Dec. 2011.

[15] G. Arslan, J. Marden, and J. Shamma, "Autonomous Vehicle-Target Assignment: A Game-Theoretical Formulation," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 129, no. 5, September 2007.

[16] R. Olfati-saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan 2007.

[17] E. Trucco and A. Verri, *Introductory techniques for 3-D computer vision*. Prentice Hall, 1998.