Title

Cooperative, Opportunistic Imaging within a Bayesian Distributed Constrained Optimization Framework

Authors

Morye, Akshay A.
Ding, Chong
Roy-Chowdhury, Amit K.
et al.

Publication Date

2012-09-14

Cooperative, Opportunistic Imaging within a Bayesian Distributed Constrained Optimization Framework

A. A. Morye, C. Ding, A. K. Roy-Chowdhury, J. A. Farrell
Department of Electrical Engineering
University of California, Riverside

***Abstract*—This article considers the design of a network of pan, tilt, zoom (PTZ) cameras to obtain high resolution facial imagery through opportunistic imaging of moving targets within a surveillance area. Camera locations are assumed to be fixed and known and the camera network is assumed to be connected. The number of targets is time-varying, often exceeding the number of cameras. The targets are free to maneuver within the surveillance area and to enter or leave the surveillance area through known doorway locations. The proposed solution involves distributed solution of a constrained, locally convex, optimization problem. The value function quantifies the per target image quality. The *tracking constraint* is an upper bound on the uncertainty in the estimated horizontal position error of each target. At each time step, optimization is over the PTZ parameters for each camera in the network with the feasible parameter space defined by the tracking constraint. All cameras optimize their PTZ parameters simultaneously by using information broadcast by neighboring cameras. At certain time steps, due to the configuration of the targets relative to the cameras, and the fact that each camera may track many targets, the camera network may be able to achieve the tracking specification with remaining degrees-of-freedom that can be used, by cameras at appropriate locations, to obtain high resolution facial images from desirable aspect angles. The challenge is to find these time instants, the appropriate camera, and the appropriate parameter settings to capitalize on them. Ideally, the system would acquire at least one high-resolution image of each target in the region before the target leaves the region. The Bayesian approach automatically trades off maximization of the objective versus risk (probability of losing track of a target). The solution involves a *Bayesian* formulation of the approach, design of the local and global objective functions and the inequality constraint set, and development of a Distributed *Lagrangian* Consensus algorithm that allows cameras to exchange information and asymptotically converge on a pair of primal-dual optimal solutions. This article presents the theoretical solution along with simulation results.**

I. Introduction

Camera sensor networks using static cameras that cover a wide-area may have low cost, but may also have low resolution imagery, may image uninteresting features or viewpoints, etc. Consequences of such scenarios are the increased difficulty of analysis of the collected video data and accumulation of vast quantities of unimportant imagery.

These issues can be addressed by employing distributed pan, tilt, zoom (PTZ) camera sensor networks. Cameras with dynamically controllable PTZ parameters can be made to differentially focus on multiple regions of interest through dynamic camera parameter selection. Such an approach would provide improved performance and greater flexibility while requiring less hardware.

A prototypical application is a security screening checkpoint at the entrance lobby of a building. Over the course of each day a high volume of people flow through the room. The room is equipped with a fixed number of cameras while the number and location of people in the room varies with time. The objective of the camera network is to track (i.e., state estimation) all persons in the room at a specified accuracy level at all times and to capture high resolution images for certain persons in the room at opportunistically selected time instants.

The challenges of such an application are development of algorithms to ensure accurate propagation of target-related information through the distributed network, analysis of the effect of changing network topology on solution convergence, and the design of objective functions suitable to solving the specified problem that also have the properties necessary to ensure convergence. All these factors influence the selection of an optimization strategy. In this paper, we formulate the problem within the Bayesian framework and utilize a distributed constrained optimization approach to compute the optimal camera parameter settings that achieve the global camera network objective through optimization of local camera objectives.

II. Literature Review

Active vision [1], [2], [3] in a camera sensor network is a research area of great interest. Active computer vision systems have been applied to applications like automatic surveillance, simultaneous localization and mapping (SLAM), trajectory planning, etc. It involves research on cooperation and coordination between many cameras in a network. Assumptions on the camera network topology play a major role in the problem solution; therefore recent work done on networks of vision sensors is important.

A recent research surveyed [4] identifies various computer vision problems that can be solved in a distributed fashion within a network topology. The article mentions the effects of nodes in the network with no measurement information and how such nodes may affect convergence properties of a distributed algorithm. An automated annealing approach for updating Lagrange multipliers to reduce dependence on agent inter-communication is provided in [5]. The method uses the probability collectives framework to generate a relation between game theory and statistical physics. The authors use

Research supported by ONR under award no.: N000140910666 for the project titled 'Distributed Dynamic Scene Analysis in a Self-Configuring Multimodal Sensor Network'.

a game-theoretic motivation to develop a parallel algorithm, but consider a non-cooperative game between agents, where the action of one agent is completely independent of the other agents in the network. Such an assumption is not appropriate for our application.

The authors of [6] proposed a game-theoretic approach to camera control, where the problem was limited to area coverage. A distributed tracking and control approach that required camera control and state-estimation to run independently and in parallel was proposed. The camera control mechanism used game-theory to assign camera settings that provided coverage over regions of interest while maintaining a high resolution shot of a target. Concurrently, a Kalman-Consensus filter [7] provided state estimate for each target. A sequential camera parameter optimization process is described and applied in [8].

Various system design methodologies are analyzed and proposed in [9]. Each agent is modeled as a self-interested decision maker within a game-theoretic environment such that it ensures attainment of local objectives for desirable global behavior of the system. The authors utilize convergence proofs within the game theory literature for a hierarchical decoupling of the global optimization problem into smaller local rules. A systematic methodology for designing agent objective functions is outlined. A similarly detailed objective function design methodology will be explained in this article.

III. Statement of Contribution

The class of PTZ camera network optimization problems considered herein is the same as that considered in [8]. The solution approach of [8] performs utility maximization sequentially and does not include an analysis of convergence. Herein, a method for distributed, cooperative and parallel optimization on a strongly connected camera sensor network is defined, analyzed, and implemented.

We redefined the camera parameter image optimization process as a *Bayesian* value function that accounts for risk arising from the uncertainty in the estimated target positions, while adhering to *Bayesian* constraints on the tracking performance. The *Bayesian* value function is designed as an ordinal potential function, such that it can be decoupled into local objectives known to each particular camera. The tracking constraint is common to all cameras.

In [10] the authors consider the distributed solution of a convex optimization problem where individual agents minimize local objectives under global inequality constraints. A general distributed constraint optimization algorithm is proposed and shown to asymptotically converge to a pair of primal-dual solutions, under certain assumptions. Our problem can be mapped to a distributed constraint optimization problem similar to that in [10]. For the case where there is only one target in the area, the local objective functions are unimodal and locally convex. However, further complexity arises from the presence of multiple targets in the area because the local objective functions become multi-modal.

IV. Problem Description and Solution Overview

The objective of the article is to develop a control mechanism for a distributed camera sensor network for opportunistic imaging of targets within a predefined surveillance area, subject to tracking constraints on targets in the area. We use the term 'opportunistic' as each camera must select its parameters to satisfy a tracking constraint at all times and to obtain high resolution facial images at times of opportunity. Such an opportunity may arise due to the high probability of image capture at a high zoom setting from a superior aspect angle, and when tracking constraints on all the targets can be simultaneously satisfied. The tracking constraints, while useful in their own right, are necessary to enable high resolution imaging.

The operating environment includes N_C cameras placed at known, fixed locations and a time-varying $N_T(t)$ number of targets with independent and unknown trajectories. It is possible that $N_T(t) > N_C$. All cameras have changeable pan ($\rho \in [-180°, 180°]$), tilt ($\tau \in [-90°, 90°]$), and zoom ($\zeta \in [\underline{F}, \bar{F}]$) parameters. We assume the cameras to have parfocal zoom lenses that maintain focus with changing focal length and have a negligible focus error.

In a distributed solution framework, each camera in the network will be required to optimize its parameters $(\rho(t_k), \tau(t_k), \zeta(t_k))$ for each imaging instant t_k, in cooperation with other cameras to maximize an objective function. The parameters of camera C_i are organized into a three vector $\mathbf{a}_i = (\rho_i, \tau_i, \zeta_i)$. Any choice of $\mathbf{a}_i$ yields a field-of-view (FoV_i) for the resulting image. That image may contain multiple targets and each target may be imaged by multiple cameras. The parameters of all cameras are organized into a vector $\mathbf{a} = [\mathbf{a}_1, \ldots, \mathbf{a}_i, \ldots, \mathbf{a}_{N_c}]$. The vector containing all parameter vectors except those of C_i will be denoted by $\mathbf{a}_{-i}$. Additional notation is summarized in Table I.

For solution of the overall problem, in the time interval $t \in (t_k, t_{k+1})$ several processes occur. Figs. 1 shows the in-

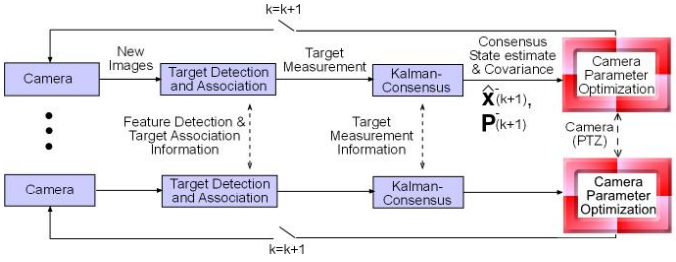

Fig. 1. Information exchange shown is only between neighboring cameras.

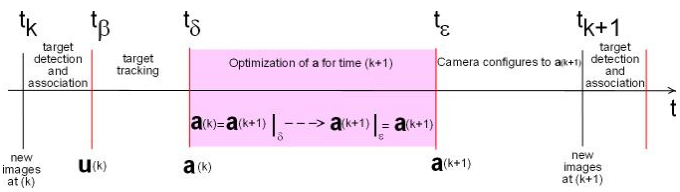

Fig. 2. Time-line of events between image sample times.

TABLE I
NOTATION SUMMARY

Parameter	Variable
Pan, Tilt, Zoom	(ρ, τ, ζ)
Min. focal length, Max. focal length	$\underline{F}$, $\bar{F}$
No. of Cameras, No. of Targets in region	N_C, $N_T(t)$
i-th camera, j-th target	C_i, T^j
(ρ, τ, ζ) settings for C_i, all cameras except C_i	$\mathbf{a}_i$, $\mathbf{a}_{-i}$
(ρ, τ, ζ) settings for all cameras	$\mathbf{a}$
Dimension of (ρ, τ, ζ) settings for C_i	$\mathbf{a}_i \in \Re^{n_i}$, $n_i = 3$
Dimension of (ρ, τ, ζ) settings for all cameras	$\mathbf{a} \in \Re^n$, $n = 3N_C$
Tracking performance vector for T^j	$\mathbf{U}_T^j(\mathbf{a})$
Tracking performance vector for all targets	$\mathbf{U}_T(\mathbf{a})$
Tracking threshold vector for all targets	$\bar{\mathbf{T}}$
Global *Bayesian* imaging value over all targets	$V_I(\mathbf{a})$
Local *Bayesian* imaging value over all targets	$V_{I_i}(\mathbf{a}_i)$
Bayesian tracking value vector for all targets	$\mathbf{V}_T(\mathbf{a})$
Lagrange multiplier vector for constraint on T^j	$\boldsymbol{\lambda}^j$
Lagrange multiplier vector for all targets	$\boldsymbol{\lambda}$
Lagrangian constructed for optimization	$L(\boldsymbol{\lambda}, \mathbf{a})$
Weight for importance of imagery of T^j by C_i	w_i^j
Image resolution obtained for T^j by C_i	$r_i^j(\mathbf{a}_i)$
Relative pose quality factor between C_i and T^j	$\alpha_i^j(\mathbf{a}_i)$
State vector for T^j	$\mathbf{x}^j$
State est., state est. covariance for T^j	$\hat{\mathbf{x}}^j$, $\mathbf{P}^j$
Fisher Information Matrix	$\mathbf{J}$
Measurement Vector, Measurement Covariance	$\mathbf{u}$, $\mathbf{C}$
Rotation Matrix from frame a to frame b	${}_a^b\mathbf{R}$
Entity e before, after new measurement	e^-, e^+
Entity e in global frame, frame defined by C_i	${}^g e$, ${}^i e$
Entity e at time-step t_k	$e(k)$
Entity e for target T^j	e^j

formation flow. Images acquired at t_k are processed for feature detection and target association. Subsequently, distributed state estimation ensures that the state of each target is estimated consistently by each camera from its and its neighbors' imagery. Consistency and accuracy of state estimation are prerequisites that enable distributed optimization of the network parameter vector $\mathbf{a}$ for high resolution image acquisition at t_{k+1}. The time sequence of processing is illustrated in Fig. 2.

Based on the previous images up to and including the image at t_k, the target state estimation process provides a prior mean $\hat{\mathbf{x}}^j(k+1)^-$ and covariance matrix $\mathbf{P}^j(k+1)^-$ for all targets (i.e., $j = 1, \ldots, N_T$). Every camera in the network has its own embedded target detection module [6], [8], [11], [12], an Extended *Kalman*-Consensus tracker [7], [13] that provides an estimate of the state of each target, and a distributed camera parameter optimizer. This article focuses on distributed camera parameter optimization.

On completion of the target state estimation process at t_δ, a prior position estimate ${}^g\hat{\mathbf{p}}^j(k+1)^-$ is available for each target T^j at the future image sampling time t_{k+1}, along with a prior covariance matrix $\mathbf{P}^j(k+1)^-$. To simplify the notation, the time argument t_{k+1} is dropped. Over the time interval $t \in (t_\delta, t_\epsilon)$, each camera will have various opportunities to optimize its parameter settings.

Remark 1: If cameras take images with a period T_s, the designer could choose to re-optimize the camera parameter settings $\mathbf{a}$ after every M-th image, resulting in $t_k = MT_s$. In this design, all M images could still be used for target detection and tracking. In this article, we choose to use $M = 1$. When $\mathbf{a}$ is not optimized for each frame, it could be interesting to choose M during operations in response to events (e.g., some target nearing the edge of the FoV, or targets nearing occlusion, etc.).

V. BACKGROUND

This section reviews optimization and game theory concepts.

A. Centralized Constrained Optimization

Consider a standard convex vector optimization (maximization) problem with a differentiable primal objective function[1] f_o and differentiable inequality constraints f^j:

$$\begin{aligned} &\text{maximize} && f_o(\mathbf{a}) \\ &\text{subject to} && f^j(\mathbf{a}) \geq 0, \quad j = 1, \cdots, m, \end{aligned} \tag{1}$$

where m is the total number of constraints and $\mathbf{a} \in \Re^n$. The *Lagrangian* $L(\boldsymbol{\lambda}; \mathbf{a})$ augments the primal objective function with the constraints:

$$L(\boldsymbol{\lambda}, \mathbf{a}) = f_o(\mathbf{a}) + \sum_{j=1}^{m} \lambda^j f^j(\mathbf{a}) = f_o(\mathbf{a}) + \boldsymbol{\lambda}^\top \mathbf{f}, \tag{2}$$

where $\mathbf{f}$ is the vector of constraint functions and $\boldsymbol{\lambda} \in \Re^m$, with $\lambda^j \geq 0$, is the *Lagrange multiplier vector*.

Since the objective and constraint functions $f_o, f^1, \cdots, f^m$ are differentiable, if an optimum $\mathbf{a}^*$ exists, then the *Lagrangian* $L(\boldsymbol{\lambda}^*, \mathbf{a}^*)$ attains its maximum at the primal-dual pair $(\mathbf{a}^*, \boldsymbol{\lambda}^*)$. The *KKT* conditions that provide a certificate for optimality are [14]:

$$\nabla f_o(\mathbf{a}^*) + \sum_{j=1}^{m} \lambda^{j*} \nabla f^j(\mathbf{a}^*) = 0, \tag{3}$$

$$f^j(\mathbf{a}^*) \geq 0, \tag{4}$$

$$\lambda^{j*} \geq 0, \tag{5}$$

$$\lambda^{j*} f^j(\mathbf{a}^*) = 0. \tag{6}$$

For the optimization problem in Eqn. (1), the primal and dual optimal points $\mathbf{a}^*$ and $\boldsymbol{\lambda}^*$ must satisfy the *KKT* conditions given by Eqns. (3 - 6).

In a centralized system, the Lagrangian is maximized by search over the parameters $\mathbf{a}$ and $\boldsymbol{\lambda}$. This requires that all data and all parameters are available at a central controller. Although, proofs of optimality are simpler and well known for this centralized approach, for reasons stated in the introduction, we are interested in decentralized solutions.

[1]Throughout this article, we use the notation such as $f_o(\mathbf{a})$ for the objective function. The notation $f_o(\mathbf{a} : \mathbf{x}^j, j = 1, \ldots, N_T)$ is more precise in making explicit the fact that the utility depends on the target state; however, it is too cumbersome to be effective.

B. Game Theory and Ordinal Potential Functions

For a distributed optimization approach, C_i will only adjust $\mathbf{a}_i$ and $\boldsymbol{\lambda}$. A challenge in formulating a distributed optimization problem is the decoupling of the system objective into local objectives, one for each agent. The game theory literature and the concept of potentiality provides guidance for addressing this challenge [15].

Consider $\mathbf{a} \in S$ and $\mathbf{a}_i,\ \mathbf{b}_i \in S_i$, where $S = S_1 \times \ldots \times S_{N_C}$ is the collection of all possible camera parameter settings in the game G, and S_i is the collection of all possible camera parameter settings for camera C_i. Let[2] $f_l(\mathbf{a}_i : \mathbf{a}_{-i})$ denote the local objective function of C_i. Game G_p is a *potential game* if there exists a potential function $\phi_p : S \mapsto \Re$ such that $\forall \mathbf{a} \in S$ and $\forall \mathbf{a}_i, \mathbf{b}_i \in S_i$,

$$\phi_p(\mathbf{b}_i, \mathbf{a}_{-i}) - \phi_p(\mathbf{a}_i, \mathbf{a}_{-i}) = f_l(\mathbf{b}_i : \mathbf{a}_{-i}) - f_l(\mathbf{a}_i : \mathbf{a}_{-i}). \quad (7)$$

Game G_o is an *ordinal potential game* if there exists an ordinal potential function $\phi_o : S \mapsto \Re$ such that $\forall \mathbf{a} \in S$ and $\forall \mathbf{a}_i, \mathbf{b}_i \in S_i$,

$$\begin{aligned} \phi_o(\mathbf{b}_i, \mathbf{a}_{-i}) - \phi_o(\mathbf{a}_i, \mathbf{a}_{-i}) &> 0 \\ \Leftrightarrow f_l(\mathbf{b}_i : \mathbf{a}_{-i}) &- f_l(\mathbf{a}_i : \mathbf{a}_{-i}) > 0. \end{aligned} \quad (8)$$

Potential games and ordinal potential games allow the global utility maximum to be achieved by maximization of the local utilities of each camera. When Eqn. (8) is satisfied, the local objective functions are said to be aligned with the global objective. Given $\phi_o(\mathbf{a})$, if the local utilities are defined as

$$f_l(\mathbf{a}_i : \mathbf{a}_{-i}) = \phi_o(\mathbf{a}_i, \mathbf{a}_{-i}), \quad (9)$$

then it is straightforward to show that the resulting game is a potential game.

Thus, by defining the global objective function as an ordinal potential function with the individual local camera objectives aligned to it, the game becomes an ordinal potential game. When the set S is compact, and a game has a continuous potential function, then the game has at least one *Nash* Equilibrium. Therefore, given any feasible initial condition, at each step for which one camera increases its own utility, the global objective function increases correspondingly, due to G being a potential game. If ϕ_o is continuous and S is compact then $\phi_o(\mathbf{a})$ is bounded above; therefore, the optimization converges toward a local maxima. At the maxima, no camera can achieve further improvement and thus a *Nash* equilibrium is reached.

C. Distributed Constrained Optimization

As described in Section V-A, to maximize the *Lagrangian* in Eqn. (2), a centralized system would find the primal-dual solution pair. For the distributed approach define the local constrained optimization problem for the i-th camera:

$$\begin{aligned} &\text{maximize} \quad f_{o_i}(\mathbf{a}_i : \mathbf{a}_{-i}) \\ &\text{subject to} \quad f^j(\mathbf{a}_i : \mathbf{a}_{-i}) \geq 0, \qquad j = 1, \cdots, m, \end{aligned} \quad (10)$$

[2]This notation $f_l(\mathbf{a}_i : \mathbf{a}_{-i})$ means that the value of the function f_l may depend on both $\mathbf{a}_i$ and $\mathbf{a}_{-i}$, but that $\mathbf{a}_i$ as treated as an independent variable while $\mathbf{a}_{-i}$ is treated as a constant by C_i.

where m is the total number of constraints and $\mathbf{a}_i \in \Re^{n_i}$. Thus the local *Lagrangian* can be formulated as,

$$\begin{aligned} L_i(\boldsymbol{\lambda}, \mathbf{a}_i : \mathbf{a}_{-i}) &= f_{o_i}(\mathbf{a}_i : \mathbf{a}_{-i}) + \sum_{j=1}^{m} \lambda^j f^j(\mathbf{a}_i : \mathbf{a}_{-i}) \\ &= f_{o_i}(\mathbf{a}_i : \mathbf{a}_{-i}) + \boldsymbol{\lambda}^\top \mathbf{f}, \end{aligned} \quad (11)$$

where $\mathbf{f}$ is the vector of constraint functions and $\boldsymbol{\lambda} \in \Re^m$ is the non-negative *Lagrange multiplier vector*.

If we define the global *Lagrangian* as the sum of local *Lagrangians*:

$$L(\boldsymbol{\lambda}, \mathbf{a}) = \sum_{i=1}^{N_C} L_i(\boldsymbol{\lambda}, \mathbf{a}_i : \mathbf{a}_{-i}), \quad (12)$$

and the objective of each agent is to maximize its local *Lagrangian*, it is straightforward to show that increasing the agent's objective leads to an increase in the global *Lagrangian*. Therefore, the local and global *Lagrangians* are aligned.

However, each agent could select a distinct value for $\boldsymbol{\lambda}$. Also, C_{i_1} adjusts $\mathbf{a}_{i_1}$, while C_{i_2} adjusts $\mathbf{a}_{i_2}$. Therefore, mechanisms are required to ensure convergence of $\mathbf{a}$ and $\boldsymbol{\lambda}$.

VI. System Model

The continuous-time state space model of target T^j is:

$$\dot{\mathbf{x}}^j(t) = \mathbf{F}\mathbf{x}^j(t) + \mathbf{G}\boldsymbol{\omega}^j(t) \quad (13)$$

where $j = 1, \ldots, N_T$ is the target number and $\mathbf{x}^j = [{}^g\mathbf{p}^j; {}^g\mathbf{v}^j]$ with ${}^g\mathbf{p}^j$ and ${}^g\mathbf{v}^j$ representing the position and velocity vectors. The process noise vector $\boldsymbol{\omega}^j \in \Re^3$ is zero mean Gaussian with power spectral density $\mathbf{Q}$.

A. Discrete-time Propagation

The discrete-time equivalent model is:

$$\mathbf{x}^j(k+1) = \boldsymbol{\Phi}\mathbf{x}^j(k) + \boldsymbol{\gamma}(k) \quad (14)$$

where $\boldsymbol{\Phi} = e^{\mathbf{F}T}$ is the state transition matrix, $\boldsymbol{\gamma} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q_d})$ is process noise, and $T = t_{k+1} - t_k$ is the sampling period.

The state estimate and its error covariance matrix are propagated between sampling instants using [16]:

$$\hat{\mathbf{x}}^j(k+1)^- = \boldsymbol{\Phi}\hat{\mathbf{x}}^j(k)^+ \quad (15)$$

$$\mathbf{P}^j(k+1)^- = \boldsymbol{\Phi}\mathbf{P}^j(k)^+\boldsymbol{\Phi}^\top + \mathbf{Q}_d. \quad (16)$$

B. Camera Coordinate Transformations

Target T^j's position in the i-th camera's frame ${}^{c_i}\mathbf{p}^j$ is related to its position in the global frame ${}^g\mathbf{p}^j$ by:

$${}^g\mathbf{p}^j = {}^g_{c_i}\mathbf{R}\ {}^{c_i}\mathbf{p}^j + {}^g\mathbf{p}_{c_i} \quad (17)$$

$${}^{c_i}\mathbf{p}^j = {}^{c_i}_g\mathbf{R}[{}^g\mathbf{p}^j - {}^g\mathbf{p}_{c_i}]. \quad (18)$$

where ${}^g\mathbf{p}_{c_i}$ is the position of C_i in global frame and ${}^{c_i}_g\mathbf{R}$ is a rotation matrix that is a function of the camera mounting angle and $\boldsymbol{a}_i$.

C. Measurement Model

Camera measurement models are derived in various references [17], so the following only presents final results for the expressions needed herein. A derivation using the same notation can be found in [8].

Let the coordinates of target T^j in the i-th camera frame be ${}^{c_i}\mathbf{p}^j = \left[{}^{c_i}x^j, {}^{c_i}y^j, {}^{c_i}z^j\right]^\top$. The standard pin-hole perspective projection camera model for camera C_i and T^j, assuming that T^j is in the FoV of C_i is,

$$ {}^{i_i}\mathbf{u}^j = \left[\begin{array}{c} \frac{(F_i)\left({}^{c_i}x^j\right)}{(s_x)\left({}^{c_i}z^j\right)} + o_x \\ \frac{(F_i)\left({}^{c_i}y^j\right)}{(s_y)\left({}^{c_i}z^j\right)} + o_y \end{array} \right] + {}^{i_i}\boldsymbol{\eta}^j, \tag{19} $$

where s_x and s_y give the effective size of a pixel in (m/pixel) measured in the horizontal and vertical directions, respectively; F_i is the focal length setting defined by $\mathbf{a}_i$; the point (o_x, o_y) gives the co-ordinates of the image plane center in pixels; and the measurement noise ${}^{i_i}\boldsymbol{\eta}^j \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_i^j)$ with $\mathbf{C}_i^j(\mathbf{a}_i) \in \Re^{2\times 2}$. The fact that the measurement noise covariance $\mathbf{C}_i^j$ is dependent on the parameter settings $\mathbf{a}_i$ is important. Note, for example, that as the focal length increases, the size of a target and the pixel uncertainty of its location in the image increase.

Given the estimated state and the camera model, the predicted measurement is

$$ {}^{i_i}\hat{\mathbf{u}}^j = \left[\begin{array}{c} \frac{(F_i)\left({}^{c_i}\hat{x}^j\right)}{(s_x)\left({}^{c_i}\hat{z}^j\right)} + o_x \\ \frac{(F_i)\left({}^{c_i}\hat{y}^j\right)}{(s_y)\left({}^{c_i}\hat{z}^j\right)} + o_y \end{array} \right]. \tag{20} $$

The measurement residual ${}^{i_i}\tilde{\mathbf{u}}^j$ is

$$ {}^{i_i}\tilde{\mathbf{u}}^j = {}^{i_i}\mathbf{u}^j - {}^{i_i}\hat{\mathbf{u}}^j. \tag{21} $$

D. Observation Matrix $\boldsymbol{H}_i^j$

The linearized relationship between the residual and the position error vector is

$$ {}^{i_i}\mathbf{u}^j - {}^{i_i}\hat{\mathbf{u}}^j \approx \mathbf{H}_i^j({}^g\mathbf{p}^j - {}^g\hat{\mathbf{p}}^j), \tag{22} $$

where $\mathbf{H}_i^j = \left.\frac{\partial {}^{i_i}\mathbf{u}^j}{\partial {}^g\mathbf{p}^j}\right|_{{}^g\hat{\mathbf{p}}^j} \in \Re^{2\times 3}$ is

$$ \mathbf{H}_i^j = \frac{F_i}{\left({}^{c_i}\hat{z}^j\right)^2}\left[\begin{array}{c} {}^g\mathbf{N}_1^{j\top} \\ {}^g\mathbf{N}_2^{j\top} \end{array} \right] = \frac{F_i}{\left({}^{c_i}\hat{z}^j\right)^2}{}^g\mathbf{N}^{j\top}, \tag{23} $$

where the matrix ${}^g\mathbf{N}^{j\top}$ is defined as:

$$ {}^g\mathbf{N}^{j\top} = \left[\begin{array}{c} {}^g\mathbf{N}_1^{j\top} \\ {}^g\mathbf{N}_2^{j\top} \end{array} \right]. \tag{24} $$

The symbol $\mathbf{N}^j$ is used as each column is normal to the vector from camera C_i's origin to the j-th target's estimated position ${}^{c_i}\hat{\mathbf{p}}^j$:

$$ {}^g\mathbf{N}_1^j = {}^g_{c_i}\mathbf{R}\,{}^{c_i}\mathbf{N}_1^j \qquad {}^{c_i}\mathbf{N}_1^j = (s_x)^{-1}\left[{}^{c_i}\hat{z}^j, 0, -{}^{c_i}\hat{x}^j\right]^\top $$
$$ {}^g\mathbf{N}_2^j = {}^g_{c_i}\mathbf{R}\,{}^{c_i}\mathbf{N}_2^j \qquad {}^{c_i}\mathbf{N}_2^j = (s_y)^{-1}\left[0, {}^{c_i}\hat{z}^j, -{}^{c_i}\hat{y}^j\right]^\top. $$

Note that ${}^g\mathbf{N}^j$ is determined by the target location relative to C_i and ${}^g_{c_i}\mathbf{R}$ is a function of $\mathbf{a}_i$. So, $\mathbf{H}_i^j$ is a function of both ${}^g\mathbf{p}^j$ and $\mathbf{a}_i$. When T^j is not in FOV_i, then $\mathbf{H}_i^j = 0 \in \Re^{2\times 3}$.

E. Measurement Update

Using Eqns. (21), (22), and (23), a measurement update for the state estimates and error covariances for targets in the area is performed using a *Kalman*-Consensus Filter described in [18].

VII. Image Optimization Methodology

The goal is to image targets at a high resolution at times of opportunity while tracking all targets at all times to a specified accuracy. The method is to perform distributed and parallel optimization to compute the optimal camera parameters $\mathbf{a}^*$ relative to Eqn. (1) at each imaging time-instant. The problem definition starts with specification of the global objective and constraints that are subsequently decoupled into local objectives for each camera. The global objective function for the constrained optimization problem is designed as a *Bayesian* imaging value function that accounts for the risk in imaging the target. Risk will be formulated using the Fisher information matrix defined as $\mathbf{J}^{j-}(k+1) = \left(\mathbf{P}^{j-}(k+1)\right)^{-1}$. The prior covariance matrix $\mathbf{P}^{j-}(k+1)$ computed using Eqn. (16) can be written in block form as[3]:

$$ \mathbf{P}^{j-} = \left[\begin{array}{cc} \mathbf{P}_{\mathbf{pp}}^{j-} & \mathbf{P}_{\mathbf{pv}}^{j-} \\ \mathbf{P}_{\mathbf{vp}}^{j-} & \mathbf{P}_{\mathbf{vv}}^{j-} \end{array} \right], \tag{25} $$

where $\mathbf{P}_{\mathbf{pp}}^{j-}$ is the prior, position error-covariance matrix.

As discussed relative to Figs. 1-2, for each image, the system detects and associates targets. These target measurements are used in a *Kalman*-Consensus Filtering approach [18] that computes ${}^g\hat{\mathbf{x}}^{j-}$ and $\mathbf{P}^{j-}$ at each camera using the model in Section VI. The selection of the camera parameters $\mathbf{a}$ affects both, the tracking accuracy and the image quality.

A. Global Optimization Problem Design

This section discusses the design and desired properties of the global imaging value function $V_I\left(\mathbf{a} : {}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right)$ and the global constraint set. The notation $V_I\left(\mathbf{a} : {}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right)$ with $j = 1, \cdots, N_T$ makes explicit that the optimization variable is $\mathbf{a}$, while the value also depends on the distribution of target T^j which is parameterized by $\left({}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right)$. For ease of notation, from this point in the paper, we will drop dependence of $V_I(\mathbf{a})$ on ${}^g\hat{\mathbf{p}}^{j-}$ and $\mathbf{P}_{\mathbf{pp}}^{j-}$, unless needed for clarity.

1) Value Function Properties: The imaging value function should have the following properties:

- **Continuously differentiable:** This is necessary for proofs of convergence, and greatly facilitates the numeric optimization process.
- **Increases with image quality:** Herein, image quality is defined by two parameters: the image resolution and the relative pose between the imaging camera and the imaged target. *Image resolution* $r_i^j(\mathbf{a}_i, {}^g\hat{\mathbf{p}}^j)$, which is a positive real number, will be quantified by the number of pixels occupied by target T^j on camera C_i's image plane. Given ${}^g\hat{\mathbf{p}}^j$, the resolution increases monotonically with zoom ζ of the imaging camera. *Relative pose* between camera

[3]The time argument $(k+1)$ is dropped for ease of notation.

C_i and target T^j will be quantified by the scalar quality factor $\alpha_i^j(\mathbf{a}_i) \in [0,1]$ defined as:

$$\alpha_i^j(\mathbf{a}_i) = \begin{cases} -(o_c\ o_o)^2 & \text{if } \mathbf{o}_{v^j} \cdot \mathbf{o}_{C_i} < 0 \\ 0 & \text{otherwise,} \end{cases} \tag{26}$$

where the scalars $o_c = \frac{\mathbf{o}_{C_i} \cdot \mathbf{o}_{T^j}}{\|\mathbf{o}_{C_i}\| \|\mathbf{o}_{T^j}\|}$, and $o_o = \frac{\mathbf{o}_{C_i} \cdot \mathbf{o}_{v^j}}{\|\mathbf{o}_{C_i}\| \|\mathbf{o}_{v^j}\|}$. The vector $\mathbf{o}_{v^j}$ is the target's direction of motion vector, and the vector $\mathbf{o}_{C_i}$ is the direction of the optical axis of the i-th camera in the global frame

$$\mathbf{o}_{C_i} =_i^g \mathbf{R}\ {}^i\mathbf{e}_3, \tag{27}$$

where $\mathbf{e}_3 = [0,0,1]^\top$. The vector $\mathbf{o}_{T^j}$ is from camera C_i's position to target T^j's estimated position. The underlying assumption is that the target is facing in the direction of its velocity vector. In Eqn. (26), the term $o_o \in [-1,1]$ yields a negative value if T^j's motion vector $\mathbf{o}_{v^j}$ is pointing towards camera C_i. The term $o_c \in [-1,1]$ yields the maximum possible positive value of 1 if C_i images target T^j such that T^j is at the center of its FoV. Hence when $\alpha_i^j \in [0,1]$ is large, it is likely that T^j is facing C_i and at the center of C_i's FoV.

- **Balanced Risk:** Risk is defined as the probability that the target is outside of the FoV of the cameras that are expected to image it. Risk increases monotonically with zoom ζ, because the area under the FoV decreases as ζ increases.

To understand the issues involved, it is informative to briefly consider the simple case where $N_T = 1$ and $\alpha_i^1 > 0$ for $i = 1, \ldots, N_C$. For this case, if risk was neglected and V_I was defined with the properties mentioned above, then each camera would maximize its focal length and select its pan and tilt parameters to center on the expected target location. If instead, the value accounted appropriately for risk, then one camera might significantly increase its zoom parameter, while the remaining cameras would use lower zoom parameters, to decrease the tracking risk due to the uncertainty in target position. The camera at the highest zoom setting would be the one at the best aspect angle.

2) Imaging Value $V_I(\boldsymbol{a} : {}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-})$: Consider the following example global objective function:

$$V_I(\mathbf{a}) = \sum_{i=1}^{N_C} \int_{\text{FoV}_i} \left(\sum_{j=1}^{N_T(t)} w^j(t)\ r_i^j(\mathbf{a}_i)\ \alpha_i^j(\mathbf{a}_i)\ p_{\mathbf{p}_j}(\mathbf{z}) \right) d\mathbf{z}, \tag{28}$$

where $w^j(t)$ is a time-varying weight that magnifies the importance of imaging certain targets relative to other targets[4]. The probability distribution $p_{\mathbf{p}_j}$ of the position of T^j in the global frame at the next imaging instant is defined herein as the Normal distribution $\mathcal{N}({}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-})$. The approach easily extends to other distributions. The dummy variable $\mathbf{z}$ representing target position is used for integration over the ground plane, where the limits of integration are provided by the i-th camera's FoV.

Each imaging camera integrates over its own FoV. The integral of image quality over FoV_i as a function of probability weighted target position yields the *Bayesian* value function, which provides the desired tradeoff between image quality and risk. The summation over the targets yields a multimodal (i.e., nonconvex) objective function when $N_T > 1$.

3) Performance Constraints: The performance constraints that ensure tracking of all targets to a specified accuracy $\bar{\mathbf{T}}$ will be defined as a function of the posterior *Fisher Information Matrix* $\mathbf{J}^{j+}\left(\mathbf{a} : {}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right)$.

- **Fisher Information:** The Fisher Information for T^j is:

$$\mathbf{J}^{j+} = \mathbf{J}^{j-} + \sum_{i=1}^{N_C} \mathbf{H}_i^{j\top} \left(\mathbf{C}_i^j\right)^{-1} \mathbf{H}_i^j \tag{29}$$

which is a function of the camera settings $\mathbf{a}$ because each $\mathbf{H}_i^j$ is a function of $\mathbf{a}_i$, as was shown in Section VI-D. In addition, $\mathbf{J}^{j+}$ depends on the distribution of target positions. The tracking utility will have to account appropriately for the probability that $T^j \in FOV_i$. Representing the posterior information $\mathbf{J}^{j+}$ in block form, we have:

$$\mathbf{J}^{j+} = \begin{bmatrix} \mathbf{J}_{\mathbf{pp}}^{j+} & \mathbf{J}_{\mathbf{pv}}^{j+} \\ \mathbf{J}_{\mathbf{vp}}^{j+} & \mathbf{J}_{\mathbf{vv}}^{j+} \end{bmatrix} \tag{30}$$

where $\mathbf{J}_{\mathbf{pp}}^{j+}$ represents the position information matrix.

- **Tracking Performance:** We define a vector $\mathbf{U}_T^j(\mathbf{a} : {}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-})$ as a measure of tracking performance for each target in the area. One example is $\mathbf{U}_T^j(\mathbf{a}) = \text{diag}\left(\mathbf{J}_{\mathbf{pp}}^{j+}\right)$. Because the quantity $\mathbf{U}_T^j(\mathbf{a})$ depends on whether $\mathbf{T}^j$ is within the FOV of each camera that is expected to image it, we define the global *Bayesian* tracking value vector $\mathbf{V}_T^j(\mathbf{a})$ as the expected value of the tracking performance vector $\mathbf{U}_T^j(\mathbf{a})$ over the position of target T^j computed across all the camera's FoVs. It is denoted as follows:

$$\mathbf{V}_T^j(\mathbf{a}) = E\left\langle \mathbf{U}_T^j(\mathbf{a}) \right\rangle = \int \left(\mathbf{U}_T^j(\mathbf{a}) p_{\mathbf{p}_j}(\mathbf{z}) \right) d\mathbf{z}, \tag{31}$$

where all variables are as defined in Eqn. (28).

- **Tracking Constraint:** The tracking constraint for each target is

$$\mathbf{V}_T^j(\mathbf{a}) \succeq \bar{\mathbf{T}}^j \tag{32}$$

where $\bar{\mathbf{T}}^j$ is a constant tracking acuracy parameter specified by the user and the notation '$\succeq$' indicates a per-element vector inequality. Stacking the *Bayesian* tracking value vectors for each target, we obtain

$$\mathbf{V}_T(\mathbf{a}) = \left[\mathbf{V}_T^1, \cdots, \mathbf{V}_T^j, \cdots, \mathbf{V}_T^{N_T} \right]^\top, \tag{33}$$

and rewrite Eqn. (32) for all targets presently in the area as:

$$\mathbf{V}_T(\mathbf{a}) \succeq \bar{\mathbf{T}}. \tag{34}$$

where $\mathbf{V}_T(\mathbf{a}), \bar{\mathbf{T}} \in \Re^{mN_T(t)}$ with $m = \dim({}^g\mathbf{p}^j)$. Eqn. (34) is the global tracking constraint.

[4]Specification of $w_i^j(t)$ is application dependent. It could be user specified or could increase as the target approaches a specified location such as an exit.

4) Global Problem Summary: The constrained global imaging value maximization problem can be written as:

$$\begin{aligned} &\text{maximize} \quad V_I\left(\mathbf{a} : {}^g\mathbf{p}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right) \\ &\text{subject to} \quad \mathbf{V}_T\left(\mathbf{a} : {}^g\mathbf{p}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right) \succeq \bar{\mathbf{T}}. \end{aligned} \tag{35}$$

The global *Lagrangian* $L(\boldsymbol{\lambda}; \mathbf{a})$ is

$$L(\boldsymbol{\lambda}; \mathbf{a}) = V_I\left(\mathbf{a}\right) + \boldsymbol{\lambda}^\top \left[\mathbf{V}_T(\mathbf{a}) - \bar{\mathbf{T}}\right], \tag{36}$$

where $L : (\boldsymbol{\lambda}, \mathbf{a}) \mapsto \Re$, $\boldsymbol{\lambda}$ is the Lagrange multiplier vector, and the sub-vector $\boldsymbol{\lambda}^j \in \Re^m$ is the Lagrange multiplier vector for the inequality constraint associated with the j-th target. Thus, to find the optimal primal-dual pair of solutions $(\mathbf{a}^*, \boldsymbol{\lambda}^*)$ through a central controller, the global unconstrained problem given by the Lagrangian in Eqn. (36) would be solved.

B. Decoupling the Global Problem

Due to the distributed nature of our problem, we need to deconstruct the global problem into smaller local problems that are solvable by each camera. The *Fisher* Information given in Eqn. (29) can be decomposed as:

$$\mathbf{J}^{j+} \quad = \quad \left[\mathbf{J}^{j-} + \mathbf{H}_{-i}^{j\top}\left(\mathbf{C}_{-i}^{j}\right)^{-1}\mathbf{H}_{-i}^{j}\right] + \mathbf{H}_{i}^{j\top}\left(\mathbf{C}_{i}^{j}\right)^{-1}\mathbf{H}_{i}^{j}.$$

While C_i is optimizing its parameters $\mathbf{a}_i$, the contribution from prior information and all other cameras (term in brackets) is considered by C_i to be constant and known. The term $\left[\mathbf{H}_{-i}^{j\top}\left(\mathbf{C}_{-i}^{j}\right)^{-1}\mathbf{H}_{-i}^{j}\right]$ is computed from $\mathbf{a}_{-i}$ which will be available through the distributed optimization process discussed in Section. VIII-C.

In our problem formulation, we allow camera C_i to optimize only its own camera parameter settings $\mathbf{a}_i$. Using this system restriction, we define the local *Bayesian* imaging value function,

$$V_{I_i}(\mathbf{a}_i) = \int_{FoV_i} \sum_{j=1}^{N_T(t)} \left(w_i^j(t)\; r_i^j(\mathbf{a}_i)\; \alpha_i^j(\mathbf{a}_i)\; p_{\mathbf{p}_j}\left(\mathbf{z}\right)\right) d\mathbf{z}. \tag{37}$$

Define $\mathbf{V}_{T_i}(\mathbf{a}_i) = \mathbf{V}_T(\mathbf{a}_i : \mathbf{a}_{-i})$ as the local constraint function for C_i. This notation concisely indicates that C_i can only alter $\mathbf{a}_i$ where for its purpose, $\mathbf{a}_{-i}$ is fixed. Of course, in general $\mathbf{V}_T(\mathbf{a}_i : \mathbf{a}_{-i}) = \mathbf{V}_T(\mathbf{a})$, so that all cameras are trying to jointly satisfy the constraints. Thus the tracking constraint for camera C_i is

$$\mathbf{V}_{T_i}(\mathbf{a}_i) \succeq \bar{\mathbf{T}}. \tag{38}$$

Note that,

$$\mathbf{V}_{T_i}(\mathbf{a}_i) \succeq \bar{\mathbf{T}} \quad \Rightarrow \quad \mathbf{V}_{T_i}^j(\mathbf{a}_i) \succeq \bar{\mathbf{T}}^j \text{ for } j = 1, \ldots, N_T(t).$$

Thus from Eqns. (29-31), we can write:

$$E\left\langle \text{diag}\left(\mathbf{J}_i^j\right)\right\rangle \succeq \bar{\mathbf{T}} - E\left\langle \text{diag}\left(\mathbf{J}^{j-} + \mathbf{J}_{-i}^j\right)\right\rangle, \tag{39}$$

where $\mathbf{J}_i^j = \mathbf{H}_i^{j\top}\left(\mathbf{C}_i^j\right)^{-1}\mathbf{H}_i^j$ and $\mathbf{J}_{-i}^j = \mathbf{H}_{-i}^{j\top}\left(\mathbf{C}_{-i}^j\right)^{-1}\mathbf{H}_{-i}^j$. Targets for which the right-hand side of Eqn. (39) is negative can be removed from the set of tracking constraints for camera C_i.

From Eqns. (37) and (38), the local imaging value maximization problem can be written as:

$$\begin{aligned} &\text{maximize} \quad V_{I_i}\left(\mathbf{a}_i : {}^g\mathbf{p}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right) \\ &\text{subject to} \quad \mathbf{V}_{T_i}\left(\mathbf{a}_i : {}^g\mathbf{p}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right) \succeq \bar{\mathbf{T}}. \end{aligned} \tag{40}$$

The local *Lagrangian* $L_i(\boldsymbol{\lambda}_i; \mathbf{a}_i)$ is:

$$L_i\left(\boldsymbol{\lambda}_i; \mathbf{a}_i\right) \quad = \quad V_{I_i}(\mathbf{a}_i) + \boldsymbol{\lambda}_i^\top \left[\mathbf{V}_{T_i}(\mathbf{a}_i) - \bar{\mathbf{T}}\right]. \tag{41}$$

Thus, for camera C_i to find its local optimal primal-dual pair of solutions $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$, C_i will maximize the local unconstrained *Lagrangian* given in Eqn. (41). It is important to note that since we assume that all cameras in the network optimize synchronously and simultaneously, the subscript i on $\boldsymbol{\lambda}_i$ in Eqn. (41) indicates that the *Lagrange* multiplier picked by camera C_i to solve the problem is a local variable and may not be globally the same throughout the network. In order to overcome this predicament, cameras in the network employ a variant of the algorithm described in [10]. This results in a consensus-step after each optimization-step[5]. The algorithm is explained in detail in Section VIII-C.

C. Lagrangian as an Ordinal Potential Function

For the problem stated in Eqn. (35), define the global objective of the multi-camera network to be the sum over the local objectives defined in Eqn. (37):

$$V_I(\mathbf{a}) \quad = \quad \sum_{i=1}^{N_C} V_{I_i}(\mathbf{a}_i). \tag{42}$$

From Eqns. (34) and (38):

$$\begin{aligned} V_T(\mathbf{a}) - \bar{\mathbf{T}} \quad &\succeq \quad \mathbf{0} \\ &\Rightarrow \quad \sum_{i=1}^{N_C}\left[V_T(\mathbf{a}) - \bar{\mathbf{T}}\right] \succeq \mathbf{0} \\ &\Rightarrow \quad \sum_{i=1}^{N_C}\left[V_{T_i}(\mathbf{a}_i) - \bar{\mathbf{T}}\right] \succeq \mathbf{0}. \end{aligned} \tag{43}$$

Thus the global *Lagrangian* in Eqn. (36) is an ordinal potential function, i.e.

$$L(\boldsymbol{\lambda}; \mathbf{a}) \quad = \quad \sum_{i=1}^{N_C} L_i(\tilde{\boldsymbol{\lambda}}; \mathbf{a}_i : \mathbf{a}_{-i}). \tag{44}$$

where $\bigcup_{i=1}^{N_C} \mathbf{a}_i = \mathbf{a}$ and $\mathbf{a}_{-i} \bigcap \mathbf{a}_i = \emptyset$. Hence, as explained in Sections V-B and V-C, Eqns. (36), (41) and (44) form an *ordinal potential game*.

[5]Every optimization and consensus step may contain multiple iterations. In fact, they may proceed simultaneously.

VIII. Distributed Lagrangian Consensus

In this section we describe a modified version of the *Distributed Lagrangian Primal-Dual Subgradient Algorithm* proposed in [10] which is used to perform average consensus on the local versions of *Lagrange* multiplier vectors.

In [10], the authors consider a convex optimization problem over a multi-agent network. The goal for the approach in [10] is for all agents to asymptotically converge to an optimal primal-dual pair under certain assumptions on network connectivity and communication. Therein, every agent maintains estimates of the primal-dual optimal pairs computed by all agents in the network. Each agent performs dynamic average consensus on the locally maintained estimates and the optimal pairs broadcast by the agent's neighbors. Consensus is performed to asymptotically converge on an optimal pair by using an update law proposed in [19].

A. Connectivity, Communication, and Consensus

To utilize the convergence properties of the algorithm described in [10], we make the following assumptions on the camera communication graph.

Assumption 1: (Connectivity) The camera communication graph is undirected, and *connected*, i.e. there exists at least one communication path from each agent to every other agent in the network.

Assumption 2: (Weights rule) There exists a scalar $\beta > 0$ such that for $C_n \in \mathcal{N}_i$, $\omega_{ii}(\kappa) \geq \beta$ and $\omega_{in}(\kappa) \in [\beta, 1]$. If cameras C_i and C_n are devoid of a communication link between them, then $\omega_{in}(\kappa) = 0$.

Remark 2: Camera C_i is only allowed to optimize $\mathbf{a}_i$. Thus, instead of performing dynamic average consensus on the optimal primal-dual solution pair, each camera need only perform dynamic average consensus on $\boldsymbol{\lambda}_i^*$ and the set of *Lagrange* multiplier vectors $\{\boldsymbol{\lambda}_n^*\}$ for $C_n \in \mathcal{N}_i$.

Remark 3: Define $B_i = C_i \bigcup \mathcal{N}_i$, from Assumptions 1 and 2, it holds that $\sum_{l \in B_i} \omega_{il}(\kappa) = 1$ and $\sum_{i \in B_i} \omega_{il}(\kappa) = 1$. This fact along with assumption 2 ensures that all cameras are *influential* [19] when cameras perform consensus on the *Lagrange* multiplier vectors.

B. Lagrange *Multiplier Update Law*

Since camera C_i receives the set of *Lagrange* multiplier vectors $\{\boldsymbol{\lambda}_n^*\}$ for $C_n \in \mathcal{N}_i$, camera C_i computes a convex combination of $\boldsymbol{\lambda}_i^*$ and $\{\boldsymbol{\lambda}_n^*\}$ as follows:

$$\boldsymbol{\nu}_{\boldsymbol{\lambda}_i}(\kappa) = \sum_{l \in B_i} \omega_{il}(\kappa)\ \boldsymbol{\lambda}_l(\kappa), \tag{45}$$

where the non-negative vector $\boldsymbol{\nu}_{\boldsymbol{\lambda}_i} \in \Re^{mN_T(t)}$ with $m = \dim({}^g\mathbf{p}^j)$. Let $\mathbf{D}_{\boldsymbol{\lambda}_i} = \nabla_{\boldsymbol{\lambda}_i} L_i(\boldsymbol{\lambda}_i(\kappa); \mathbf{a}_i^*)$ be the gradient of the local *Lagrangian* with respect to $\boldsymbol{\lambda}_i$. Considering that assumptions 1 and 2 hold, camera C_i performs a *Lagrange* multiplier update using the following update law:

$$\boldsymbol{\lambda}_i(\kappa + 1) = \boldsymbol{\Theta}_{M_i} \left[\boldsymbol{\nu}_{\boldsymbol{\lambda}_i}(\kappa) + s(\kappa)\ \mathbf{D}_{\boldsymbol{\lambda}_i}^{\top}(\kappa) \right], \tag{46}$$

where $\mathbf{D}_{\boldsymbol{\lambda}_i} \in \Re^{mN_T(t)}$, and the scalar $s(\kappa) > 0$ is the step-size. The matrix $\boldsymbol{\Theta}_{M_i} \in \Re^{mN_T(t) \times mN_T(t)}$ is an alternating projection operator [20]. The projection operator is used to project the *Lagrange* multiplier vector into the set M_i which contains all the dual optimal solutions for C_i. It is shown in [10], [19] that for all nodes,

$$\lim_{\kappa \to \infty} \|\boldsymbol{\lambda}_c - \boldsymbol{\lambda}_i(\kappa)\| = 0. \tag{47}$$

The convexity of M_i and convergence properties of $\boldsymbol{\Theta}_{M_i}$ are described in Appendix A and [19].

C. Optimization

Since the optimization problem described by Eqn. (40) is non-convex, any solution found may only be locally optimal. The distributed optimization process can be broken down into three separate steps:

1) **Camera Parameter Optimization:** Camera C_i selects $(\mathbf{a}_i, \boldsymbol{\lambda}_i)$ to maximize $L_i(\boldsymbol{\lambda}_i; \mathbf{a}_i)$ while holding $\mathbf{a}_{-i}$ constant. It then communicates the newly computed local primal-dual pair $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$ and $\mathbf{a}_{-i}$ to its set of neighbors denoted by $\mathcal{N}_i$. Parameter broadcast rule is described in Section VIII-D.
2) **Camera Parameter Replacement:** If camera $C_n \in \mathcal{N}_i$, then C_n receives $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*, \mathbf{a}_{-i})$ and replaces its previous value of $(\mathbf{a}_i, \mathbf{a}_{-i})$ with $(\mathbf{a}_i^*, \mathbf{a}_{-i})$. Similarly, camera C_i receives $(\mathbf{a}_n^*, \boldsymbol{\lambda}_n^*, \mathbf{a}_{-n})$ and performs the replacement step on $(\mathbf{a}_n, \mathbf{a}_{-n})$. The rules of replacement are described in Section VIII-D.
3) **Consensus on Lagrange Multipliers:** Using the *Lagrange* multiplier update law in Eqn. (46), C_n performs dynamic average consensus on its local *Lagrange* multiplier vector $\boldsymbol{\lambda}_n^*$ and the *Lagrange* multiplier vectors received from cameras in $\mathcal{N}_n$ to converge toward the consensus *Lagrange* multiplier vector $\boldsymbol{\lambda}_c$.

This distributed optimization process is then repeated.

D. Camera Parameter Replacement Rule

After C_i computes $\mathbf{a}_i^*$, let $\mathbf{a}^i \in \Re^{3N_C}$ be camera C_i's version of the camera parameter vector $\mathbf{a}$, such that $\mathbf{a}^i = \left[\mathbf{a}_1^i, \cdots, \mathbf{a}_i^*, \cdots, \mathbf{a}_{N_C}^i\right]^{\top}$. Camera C_i broadcasts $\mathbf{a}_i^*$ to its neighbors in $\mathcal{N}_i$ along with the new parameters $\mathbf{a}_l^i \in \Re^3$ that it received this iteration. New parameters are defined as follows. The camera parameter vector $\mathbf{a}_l^i$ of camera C_l available at camera C_i as a sub-vector[6] of $\mathbf{a}^i$ was received at a known time-stamp κ_l^i. When a neighbor $C_n \in \mathcal{N}_i$ communicates a camera parameter vector $\mathbf{a}_l^n$, time-stamped as valid at κ_l^n, camera C_i uses the following rule for replacement:

$$\mathbf{a}_l^i = \begin{cases} \mathbf{a}_l^i\left(\kappa_l^i\right) & \text{if } \kappa_l^i \geq \kappa_l^n \\ \mathbf{a}_l^n\left(\kappa_l^n\right) & \text{otherwise.} \end{cases} \tag{48}$$

Eqn. (48) results in a local replacement of only those camera parameters that are not as recent as the parameters received from the set of neighbors.

After replacement, C_i updates the local time-stamps, computes $\boldsymbol{\lambda}_c$, and repeats the local optimization process at iteration

[6]Note that C_l may or may not be in the set $\mathcal{N}_i$ as $\mathbf{a}_l^i$ could have propagated through the network to C_i over time.

step κ_o. It then broadcasts $\mathbf{a}_i^*(\kappa_o)$, $\boldsymbol{\lambda}_i^*(\kappa_o)$, and only the most recently updated camera parameter sub-vectors from $\mathbf{a}_{-i}$.

Thus, by using the replacement step in Eqn. (48), and the *Lagrange* multiplier update law from Eqn. (46), at each consensus iteration[7] κ, every camera maintains an estimate of the primal-dual pairs of all cameras, which are the saddle-points of the local *Lagrangians*.

E. Certificate for Optimality

For each unconstrained maximization problem given by Eqn. (41), the *KKT* conditions [14] are:

$$\nabla V_{I_i}(\mathbf{a}_i^* :) + \left[\nabla \mathbf{V}_{T_i}(\mathbf{a}_i^*)\right]^\top \boldsymbol{\lambda}_i^* = 0, \quad (49)$$

$$\mathbf{V}_{T_i}(\mathbf{a}_i^*) - \bar{\mathbf{T}} \succeq \mathbf{0}, \quad (50)$$

$$\boldsymbol{\lambda}_i^* \succeq \mathbf{0}, \quad (51)$$

$$\boldsymbol{\lambda}_i^{*\top} \left[\mathbf{V}_{T_i}(\mathbf{a}_i^*) - \bar{\mathbf{T}}\right] = 0, \quad (52)$$

where, for the optimization problem in Eqn. (40), the optimal primal-dual pair $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$ must satisfy the *KKT* conditions given by Eqns. (49 - 52).

It must be noted that all cameras optimize in parallel and broadcast $\mathbf{a}_i^*$, $\mathbf{a}_{-i}$ and $\boldsymbol{\lambda}_i^*$ to their neighbors which propagate them through the network. While any camera is locally optimizing its settings, it is accounting for the prior information, an updated $\boldsymbol{\lambda}_c$, and the currently best settings of all the other cameras.

For the solution approach described in [8], optimization stops when either an optimum is achieved, a user-defined heuristic as a stopping condition is met, or as shown in Fig 2, the time interval $t \in [t_\delta, t_\epsilon]$ allotted for optimization elapses. In case the system fails in achieving the optimum, there is no guarantee of the solution computed being feasible. Whereas, the *KKT* conditions described in Eqns. (49 - 52) provide a certificate on optimality of the solution. A non-heuristic stopping condition on optimality is provided by the *Lagrangian* L_i at the primal-dual optimal pair $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$ as $\nabla L_i\left(\boldsymbol{\lambda}_i^*; \mathbf{a}_i^*\right) = 0$. Thus, when the time interval $t \in [t_\delta, t_\epsilon]$ allotted for optimization elapses, even if the solution is sub-optimal, the solution obtained is guaranteed to be feasible. This results in all targets being tracked to the specified tracking accuracy at all times, while procuring high-resolution imagery when opportunity arises. After optimization, cameras reconfigure themselves in the time interval $t \in [t_\epsilon, t_{k+1}]$, in readiness for upcoming images at t_{k+1}.

IX. Implementation

This section describes an implementation of the procedure proposed in this article, implemented in a simulation in *Matlab*. The goal of the simulation is to show that a distributed network of cameras obtains opportunistic high-resolution facial imagery of targets in the area, while tracking all targets at all times to a specified tracking accuracy.

[7]Note that at the first update iteration, all cameras use the optimum primal-dual pair $(\mathbf{a}^*, \boldsymbol{\lambda}_c)$.

A. Scenario and Setup

As shown in Fig. 4, we set up an area of $400\ m^2$ being monitored by $N_C = 3$ calibrated cameras with the following position vectors in meters:

$$C_1 = [10, 0, 4]^\top, \quad C_2 = [0, 10, 4]^\top, \quad C_3 = [20, 10, 4]^\top.$$

The FoVs of each individual camera are color-coordinated with its colored position marker. To emulate a real-life setup, it is imperative that the entrance of the area be constantly monitored. This is done by treating it as a stationary target with an initial covariance that considers the width of the entrance. This is done so that the camera network detects and has an accurate initial state estimate for targets entering the area. While there is tracking value to be gained by imaging the entrance, the imaging value gained is always zero.

B. Experiment Details

The simulation was run for $T = 30$ seconds, with target state propagating through time. Targets enter the area from the entrance at random times. Thus the total number of targets in the area is time variant. To make results concise and understandable, the maximum number of targets permissible in the area was limited to $\overline{N}_T = 6$ targets, where $0 \leq N_T(t) \leq \overline{N}_T$. After entering the area, each target was able to travel on a random trajectory.

1) Optimization: All cameras optimize contemporaneously. The network of cameras were required to satisfy the tracking constraint in Eqn. (38), where $\bar{\mathbf{T}} = 1\mathrm{m}^{-2}$. Camera C_i receives camera parameters $\mathbf{a}_{-i}^*$ from its neighbors, and uses its current parameters $\mathbf{a}_i$ to compute Eqn. (29) and then computes $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$ through maximization of Eqn. (41). The optimum $\mathbf{a}_i^*$ is bounded by permissible camera specifications[8] $\underline{\mathbf{a}}_i$ and $\overline{\mathbf{a}}_i$ The specifications $\underline{\mathbf{a}}_i$ and $\overline{\mathbf{a}}_i$ are the lower and upper bounds for possible values of $\mathbf{a}_i$ respectively. We use an interior-point method [21] to perform optimization.

C. Results

Each target enters the area at a random time-instant that is unknown to any of the cameras in the network. While in the area, targets adopt the motion model in Eqn. (14). On hitting any wall of the area, the target bounces and stays within the area.

For this run of the simulation, targets T^1 to T^6 entered the area at times $1, 3.53, 4.25, 4.86, 5.09,$ and 8.61, respectively. As each T^j enters, a camera monitoring the entrance detects and images it at imaging time. By the time the simulation ends, all targets were active in the room and approaching the exit.

1) Bayesian Imaging and Tracking Performance: Cameras try to maximize their local *Lagrangians*[9] $L_i(\boldsymbol{\lambda}_c; \mathbf{a}_i^*)$ by satisfying the tracking spec and maximizing the local *Bayesian* imaging values. Fig. 3a shows that the planned *Bayesian*

[8]These specifications do not adhere to any specific camera configuration and have been chosen for the sake of implementation.

[9]Note that $\boldsymbol{\lambda}_c$ is obtained from $\{\boldsymbol{\lambda}_i^*, \boldsymbol{\lambda}_{-i}^*\}$ using the algorithm described in Section VIII.

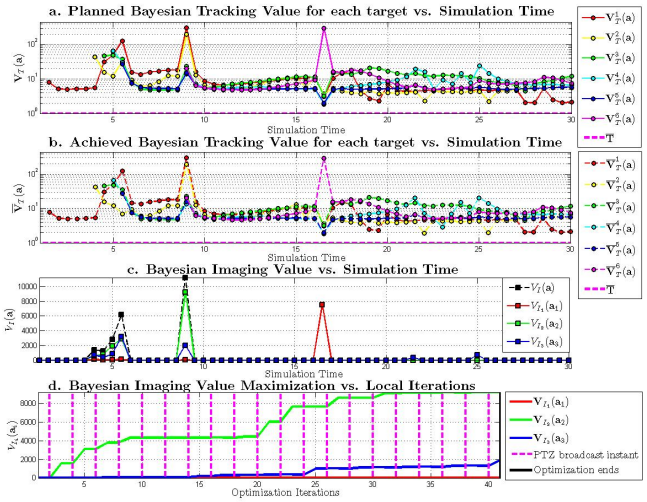

Fig. 3. *Bayesian* Tracking and Imaging Values: Fig. a (top) shows that the camera network successfully and co-operatively satisfies the tracking threshold $\bar{\mathbf{T}}$ on every target, at all times. This results in feasible optimum solutions . Fig. b (second from top) shows the achieved tracking value by the network. Random motion of targets may result in this value not satisfying the spec. In Fig. c (second from bottom), a high resolution facial image capture by camera C_i is indicated by a spike for the local *Bayesian* Imaging Value function $V_{I_i}(\mathbf{a}_i)$. The plot also shows the global *Bayesian* Imaging Value $V_I(\mathbf{a})$ as a sum of the local values. Cameras obtain high-res facial shots of targets at various times of opportunity. Fig. d (bottom) shows the local optimization iterations of the cameras at time-instant t_9. Within the time allotted for cameras to optimize, cameras achieve a local optimum and broadcast their PTZ parameters to update their local estimates of $(\mathbf{a}, \boldsymbol{\lambda})$. Local optimization resumes after solutions are updated, and the process repeats.

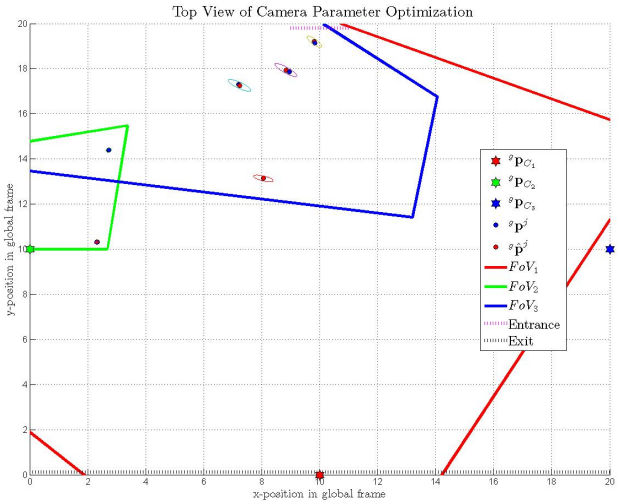

Fig. 4. Top-view of Opportunistic High-res Facial Image Capture: The figure is a plot of the optimized FoVs of the cameras after a feasible optimal solution is achieved at time-instant t_9. Cameras start with a wide FoV before optimization begins. As shown in Fig. 3, camera C_2 obtains high imaging values while all cameras cooperatively satisfy the tracking constraint.

tracking value $V_T^j(\mathbf{a}^*)$ maintained by the network of cameras on every target in the area is always greater than the tracking spec, at all times. Also, from Fig. 3a, it is clear that all primal-dual $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$ solutions obtained through local optimization are within the feasible set. From Eqn. (52), it is thus trivial to prove that the *Lagrange* multiplier vectors $\boldsymbol{\lambda}_i^*(t)$ for all cameras are $\mathbf{0} \in \Re^{2N_T}$.

The network cooperatively maintains track on targets, by predicting their motion based on a-priori information. But an unexpected maneuver by a target, can lead to a drop in accuracy. Fig. 3b shows the tracking value achieved by the network by tracking every target. A compromise on the achieved tracking value may occur on account of the random motion of targets. Such an occurrence can be seen in the case of tracking target T^2 at time-step t_6.

Fig. 3c shows cameras procuring opportunistic high resolution facial images of targets, when tracking constraints on all targets are satisfied. A high value for $V_{I_i}(\mathbf{a}_i^*)$ indicates a high resolution facial capture by camera C_i. For the initial portion of the displayed run of the simulation, target dynamics dictated that cameras C_2 and C_3 were more suited for high-res facial capture. Camera C_1 obtains a high-res facial image at time-step t_{18}.

For this simulation, the local dynamic imaging weight $w_i^j(t)$ on camera C_i imaging target T^j is defined as a continuously differentiable and bounded function:

$$w_i^j(t) = \left(1 + \exp\left[l_1\left[\bar{V} - \exp\left[l_2\left[\bar{d}^j - d_e^j(t)\right]\right] V_{I_i}^j(t)\right]\right]\right)^{-1}, \tag{53}$$

where $\bar{V} = \max(V_{I_i}^j(1), \cdots, V_{I_i}^j(t-1))$, $\bar{d}^j$ is the known distance between the area exit and T^j's position of entrance, and $d_e^j(t)$ is the distance between the area exit and T^j's position at time t. Such a choice of $w_i^j(t)$, for large l_1 and l_2, ensures that the maximization of $w_i^j(t)$ for imaging target T^j is only factored in if the imaging value $V_{I_i}^j(t)$ obtained by C_i exceeds the previous best imaging value specified by $\bar{V}$ or if the target is approaching the exit. This is done to prioritize imaging of other targets in the area if a better image of T^j has previously been captured by any camera, and also to provide a higher weight on imaging T^j if it is approaching the exit and has not yet been captured with sufficient quality.

Fig. 3d is an illustration of the sub-process that all cameras perform to provide co-operative solutions at time-instant t_9. As shown in the figure, all cameras reach a local optimum before broadcasting their primal-dual solutions, updating their local estimates of $(\mathbf{a}, \boldsymbol{\lambda})$ and resuming the optimization process. This process repeats till an optimum is reached or time for optimization comes to an end.

Remark 4: Cameras may or may not require an equal number of iterations to reach an optimum. Cameras were assigned a maximum of 20 PTZ broadcast[10] instances. Note that due to the cooperative nature of the solution, after broadcast of $\mathbf{a}_i$, there may or may not be an increase in the local Imaging Value.

Fig. 5 shows that the network of PTZ cameras obtain at least one high resolution facial capture of every target in

[10]Cameras complete camera parameter optimization before the broadcast instant while performing parameter replacement and *Lagrange* multiplier consensus after broadcast.

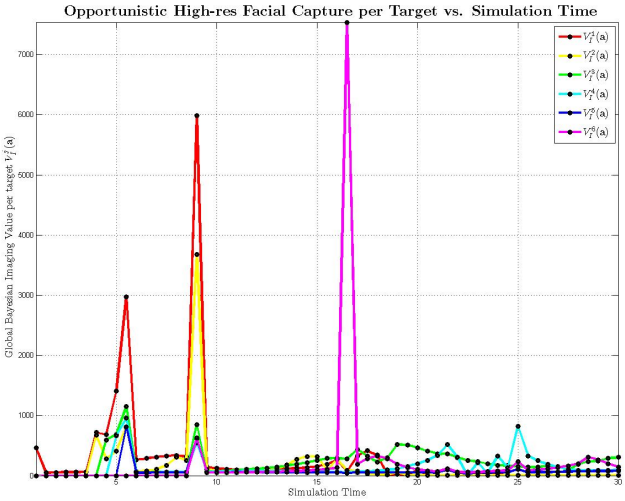

Fig. 5. Opportunistic High-res Facial Image Capture: The figure shows that the camera network obtains at least one high-res facial image of every target in the area, at opportunistic times. A high resolution facial image capture of T^j is indicated by a spike for the global *Bayesian* Imaging Value function $V_I(\mathbf{a})$. The times of opportunity are dictated by the value to be gained by imaging T^j and by the cooperative behavior of the cameras to satisfy the tracking spec. Take the case of target T^1 for example. After T^1 enters the area at time t_1, as shown in Fig. 3c, camera C_1 images it to satisfy the tracking constraints. As T^1 moves through the room, at various times, cameras have opportunities to image it.

the area. The local dynamic imaging weight in Eqn. (53) ensures that the capture of a high-res facial image occurs if the opportunity for an image of higher quality than the previous capture becomes available. Eqn. (53) also ensures high imaging weights to targets approaching the exit. A camera's action of zooming in to take the image is countered by random entrances of other targets. As other targets enter the area, cameras are compelled to image the incoming targets. This restricts the number of chances cameras get to zoom in and image targets.

At the end of the simulation, it can be concluded that all targets were always tracked to an accuracy better than $\bar{\mathbf{T}}$, and when an opportunity arose (constraint was satisfied and facial imagery was possible), the network responded cooperatively with high resolution facial images of targets based on target dynamics.

X. Conclusion and Future Work

In this article, we propose a method for a network of distributed cameras to opportunistically obtain high-resolution images, while always tracking all targets at all times. Distributed Constrained Optimization to image and track targets within a *Bayesian* framework is presented. This guarantees a feasible solution for targets maneuvering with random trajectories, even if the solution obtained is sub-optimal. Moreover, a *Lagrangian* consensus algorithm based on a primal-dual optimization method is proposed to perform distributed, cooperative and contemporaneous optimization across all cameras in the network.

The global optimization problem is formulated as a potential game with the global objective decoupled as smaller local problems with tracking constraints on all targets. This formulation enables using existing convergence proofs available in game-theory, thus making this framework resourceful. Depending on the target's position, orientation and direction of motion, an image resolution and relative pose quality factor describing the quality of image captured by the cameras is included in the *Bayesian* imaging value function. A dynamic weighting scheme for targets is utilized, since the importance of a target reduces drastically once its high resolution image has been captured by the network, unless a better aspect angle and facial resolution for capture can be obtained.

Designing a value function that considers a planning horizon for the dynamic panning, tilting and zooming of cameras to enforce continuity of optimum parameters versus time, is critical to minimize mechanical wear of the camera and provide a smooth video feed. Dynamic-programming or receding horizon control based approaches are worth investigating. Value functions based on target activity is another intriguing facet that can be explored, along with exploration of other image quality-oriented functions. Implementation of the proposed approach on a Camera Network test-bed is in process.

Appendix

A. Convexity of the Dual Optimal Set

The projection operator $\mathbf{\Theta}_{M_i}$ of Section VIII-B requires a compact set M_i. In this section we follow the approach of [10] to compute such a compact set M_i containing all the dual-optimal solutions. Corresponding to the notations of [10], we rewrite the global imaging value maximization problem in Eqn. (35) as a global cost minimization problem in the following manner:

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{a}) \\ &\text{subject to} \quad \mathbf{g}(\mathbf{a}) \preceq \mathbf{0}, \end{aligned} \tag{54}$$

where the global objective $f : \Re^n \rightarrow \Re$ is defined as $f(\mathbf{a}) = -V_I(\mathbf{a})$ and the vector of constraints $\mathbf{g} : \Re^n \rightarrow \Re^{2N_T}$ is defined as $\mathbf{g}(\mathbf{a}) = \bar{\mathbf{T}} - \mathbf{V}_T(\mathbf{a})$. As mentioned in Section V-B, $\mathbf{a} \in S$ and $\mathbf{a}_i \in S_i$, where $S = S_1 \times \ldots \times S_{N_C}$ is the collection of all possible camera parameter settings, and S_i is the collection of all possible camera parameter settings for camera C_i.

The dual of the optimization problem in Eqn. (54) is

$$\begin{aligned} &\text{maximize} \quad q(\boldsymbol{\lambda}) \\ &\text{subject to} \quad \boldsymbol{\lambda} \succeq \mathbf{0}, \end{aligned} \tag{55}$$

where $q(\boldsymbol{\lambda}) = \inf_{\mathbf{a} \in S}\{f(\mathbf{a}) + \boldsymbol{\lambda}^\top \mathbf{g}(\mathbf{a})\} : \Re^{2N_T}_{\succeq \mathbf{0}} \rightarrow \Re$ is the dual function of $f(\mathbf{a})$.

Let $\mathbf{a}^*$ be the optimal value of the global minimization problem given in Eqn. (54) and $\mathbf{a}^* = [\mathbf{a}_1^*, \ldots, \mathbf{a}_{N_C}^*]^\top$. Let $f^* = f(\mathbf{a}^*)$ and $q^* = q(\boldsymbol{\lambda}^*)$ be the function values at the global primal optimal point $\mathbf{a}^*$ and the dual optimal point $\boldsymbol{\lambda}^*$,

respectively. From the property of weak duality $f^* \geq q^*$. This implies that $\forall(\mathbf{a}, \boldsymbol{\lambda}) \in S \times \Re^{2N_T}_{\succeq \mathbf{0}}$,

$$f(\mathbf{a}) \geq q(\boldsymbol{\lambda}). \tag{56}$$

From Eqns. (42) and (43), we can write the global objective as:

$$f(\mathbf{a}) \quad = \quad \sum_{i=1}^{N_C} f_i(\mathbf{a}_i), \tag{57}$$

and the global constraints as:

$$\mathbf{g}(\mathbf{a}) \quad = \quad \sum_{i=1}^{N_C} \mathbf{g}_i(\mathbf{a}_i : \mathbf{a}_{-i}), \tag{58}$$

where the functions $f_i : \Re^{n_i} \rightarrow \Re$ and $\mathbf{g}_i : \Re^{n_i} \rightarrow \Re^{2N_T}$ are defined as $f_i(\mathbf{a}_i) = -V_{I_i}(\mathbf{a}_i)$ and $\mathbf{g}_i(\mathbf{a}_i : \mathbf{a}_{-i}) = \bar{\mathbf{T}} - \mathbf{V}_{T_i}(\mathbf{a}_i)$. The local cost minimization problem is

$$\begin{aligned} &\text{minimize} \quad f_i(\mathbf{a}_i) \\ &\text{subject to} \quad \mathbf{g}_i(\mathbf{a}_i : \mathbf{a}_{-i}) \preceq \mathbf{0}. \end{aligned} \tag{59}$$

The dual of local optimization problem in Eqn. (59) is

$$\begin{aligned} &\text{maximize} \quad q_i(\boldsymbol{\lambda}) \\ &\text{subject to} \quad \boldsymbol{\lambda} \succeq \mathbf{0}, \end{aligned} \tag{60}$$

where $q_i(\boldsymbol{\lambda}) = \inf_{\mathbf{a}_i \in S_i} \{f_i(\mathbf{a}_i) + \boldsymbol{\lambda}^\top \mathbf{g}_i(\mathbf{a}_i : \mathbf{a}_{-i})\} : \Re^{2N_T}_{\succeq \mathbf{0}} \rightarrow \Re$ is the dual function of $f_i(\mathbf{a}_i)$.

Let $f_i^* = f_i(\mathbf{a}_i^*)$ and $q_i^* = q_i(\boldsymbol{\lambda}^*)$ be the function values at the local primal optimal point $\mathbf{a}_i^*$ and the dual optimal point $\boldsymbol{\lambda}^*$, respectively. From the property of weak duality $f_i^* \geq q_i^*$. This implies that $\forall(\mathbf{a}_i, \boldsymbol{\lambda}) \in S_i \times \Re^{2N_T}_{\succeq \mathbf{0}}$,

$$f_i(\mathbf{a}_i) \geq q_i(\boldsymbol{\lambda}). \tag{61}$$

It should be noted that for all $\boldsymbol{\lambda} \succeq \mathbf{0}$,

$$\begin{aligned} q\left(\boldsymbol{\lambda}\right) \quad &= \quad \inf_{\mathbf{a} \in A} \left\{ \sum_{i=1}^{N_C} \left(f_i(\mathbf{a}_i) + \boldsymbol{\lambda}^\top \mathbf{g}_i(\mathbf{a}_i : \mathbf{a}_{-i}) \right) \right\} \\ &\geq \quad \sum_{i=1}^{N_C} \inf_{\mathbf{a}_i \in A_i} \left\{ f_i(\mathbf{a}_i) + \boldsymbol{\lambda}^\top \mathbf{g}_i(\mathbf{a}_i : \mathbf{a}_{-i}) \right\} \\ q\left(\boldsymbol{\lambda}\right) \quad &\geq \quad \sum_{i=1}^{N_C} q_i(\boldsymbol{\lambda}). \end{aligned} \tag{62}$$

Thus, from Eqns. (56), (57) and (62), we can write

$$\sum_{i=1}^{N_C} f_i(\mathbf{a}_i) \geq q(\boldsymbol{\lambda}) \geq \sum_{i=1}^{N_C} q_i(\boldsymbol{\lambda}). \tag{63}$$

Assume that there exists a vector $\bar{\mathbf{a}} \in \Re^n$ such that $\mathbf{g}(\bar{\mathbf{a}}) \prec \mathbf{0}$, i.e. *Slater's* condition is satisfied. Then, $\forall \bar{\boldsymbol{\lambda}} \succeq \mathbf{0}$, define the non-empty set

$$Q_{\bar{\boldsymbol{\lambda}}} = \{\boldsymbol{\lambda} \succeq \mathbf{0} \mid q\left(\boldsymbol{\lambda}\right) \geq q\left(\bar{\boldsymbol{\lambda}}\right)\}. \tag{64}$$

Lemma 1 in [19] guarantees that the set $Q_{\bar{\boldsymbol{\lambda}}}$ is bounded as:

$$\max_{\boldsymbol{\lambda} \in Q_{\bar{\boldsymbol{\lambda}}}} \parallel \boldsymbol{\lambda} \parallel \quad \leq \quad \frac{1}{\xi(\bar{\mathbf{a}})} \left(f\left(\bar{\mathbf{a}}\right) - q\left(\bar{\boldsymbol{\lambda}}\right) \right), \tag{65}$$

where $\xi(\bar{\mathbf{a}}) = \min_{1 \leq l \leq 2N_T} \{-g^l(\bar{\mathbf{a}})\}$, and g^l is the l-th element of $\mathbf{g} \in \Re^{2N_T}$. Define the set of dual optimal points Q^* as:

$$Q^* = \{\boldsymbol{\lambda} \succeq \mathbf{0} \mid q(\boldsymbol{\lambda}) \geq q^*\}. \tag{66}$$

Note that $Q^* \subseteq Q_{\bar{\boldsymbol{\lambda}}}$. Hence, from Eqn. (65)

$$\max_{\boldsymbol{\lambda}^* \in Q^*} \parallel \boldsymbol{\lambda}^* \parallel \quad \leq \quad \max_{\boldsymbol{\lambda} \in Q_{\bar{\boldsymbol{\lambda}}}} \parallel \boldsymbol{\lambda} \parallel \quad \leq \quad \frac{1}{\xi(\bar{\mathbf{a}})} \left(f\left(\bar{\mathbf{a}}\right) - q^* \right). \tag{67}$$

We need to show that the set of dual optimal points Q^* is contained by a non-empty, convex, and compact set. For this purpose, we define the function $r_i : S \times \Re^{2N_T}_{\succeq \mathbf{0}} \mapsto \Re$ as[11]:

$$r_i(\mathbf{a}_i; \boldsymbol{\lambda} : \mathbf{a}_{-i}) = \frac{1}{\xi_i(\mathbf{a}_i : \mathbf{a}_{-i})} \sum_{i=1}^{N_C} \left(f_i\left(\mathbf{a}_i\right) - q_i\left(\boldsymbol{\lambda}\right) \right). \tag{68}$$

In Eqn. (68), the scalar $\xi_i(\bar{\mathbf{a}}_i : \bar{\mathbf{a}}_{-i}) = \xi(\bar{\mathbf{a}})$ is given as:

$$\xi_i(\bar{\mathbf{a}}_i : \bar{\mathbf{a}}_{-i}) = \min_{1 \leq l \leq 2N_T} \left\{ - \sum_{m=1}^{N_C} g_m^l(\bar{\mathbf{a}}_m : \bar{\mathbf{a}}_{-m}) \right\}, \tag{69}$$

where g_m^l is the l-th element of $\mathbf{g}_m \in \Re^{2N_T}$.

For the *Slater* vector $\bar{\mathbf{a}}$, we can write $\xi(\bar{\mathbf{a}}) = \xi_i(\bar{\mathbf{a}}_i : \bar{\mathbf{a}}_{-i}) > 0$. This, along with Eqn. (61), implies that $\forall \boldsymbol{\lambda} \succeq \mathbf{0}$, the function $r_i(\bar{\mathbf{a}}_i; \boldsymbol{\lambda} : \bar{\mathbf{a}}_{-i}) \geq 0$. Thus, by picking any $\hat{\boldsymbol{\lambda}} \succeq \mathbf{0}$ and a positive scalar θ_i, we can define the following non-empty, convex, and compact set:

$$M_i(\bar{\mathbf{a}}_i, \hat{\boldsymbol{\lambda}} : \bar{\mathbf{a}}_{-i}) = \{\boldsymbol{\lambda} \succeq \mathbf{0} \mid\mid \parallel \boldsymbol{\lambda} \parallel \leq r_i(\bar{\mathbf{a}}_i; \boldsymbol{\lambda} : \bar{\mathbf{a}}_{-i}) + \theta_i\}. \tag{70}$$

From Eqns. (62) and (67), we can write:

$$\begin{aligned} \max_{\boldsymbol{\lambda}^* \in Q^*} \parallel \boldsymbol{\lambda}^* \parallel \quad &\leq \quad \frac{1}{\xi(\bar{\mathbf{a}})} \left(f(\bar{\mathbf{a}}) - q(\hat{\boldsymbol{\lambda}}) \right) \\ &\leq \quad \frac{1}{\xi_i(\bar{\mathbf{a}}_i : \bar{\mathbf{a}}_{-i})} \sum_{i=1}^{N_C} \left(f_i(\bar{\mathbf{a}}_i) - q_i(\hat{\boldsymbol{\lambda}}) \right) \\ &= \quad r_i\left(\bar{\mathbf{a}}_i; \hat{\boldsymbol{\lambda}} : \bar{\mathbf{a}}_{-i} \right). \end{aligned} \tag{71}$$

Thus the dual optimal set $Q^* \subseteq M_i(\bar{\mathbf{a}}_i, \hat{\boldsymbol{\lambda}} : \bar{\mathbf{a}}_{-i})$ for every camera in the network.

Based on the imaging value function properties and function definition discussed in Section VII-A, the global objective function $f(\mathbf{a})$ and the local objective function $f_i(\mathbf{a}_i)$ are continuous and locally convex. The global dual $q(\boldsymbol{\lambda})$ of the global objective function $f(\mathbf{a})$, and the local dual $q_i(\boldsymbol{\lambda})$ of the local objective function $f_i(\mathbf{a}_i)$ are also continuous and locally convex. From Eqns. (68) and (69), the function r_i is also continuous, and locally convex. Hence the set M_i is non-empty, locally convex, and locally compact. Thus an alternating projection operator $\boldsymbol{\Theta}_{M_i}$ can be used to project the computed dual points into a locally compact set M_i. A method to compute $M_i(\bar{\mathbf{a}}_i, \hat{\boldsymbol{\lambda}} : \bar{\mathbf{a}}_{-i})$ is proposed in [10].

[11]Note that r_i is a function of $\mathbf{a} \in S$, where $S = S_1 \times \ldots \times S_{N_C}$, and $\boldsymbol{\lambda} \succeq \mathbf{0}$. Since in our distributed approach, camera C_i can only alter $\mathbf{a}_i$, we can write r_i as a function of $\mathbf{a}_i$ and $\boldsymbol{\lambda}$ given $\mathbf{a}_{-i}$.

REFERENCES

[1] Aloimonos, Y. and Weiss, I. and Bandyopadhyay, A., "Active Vision," in *IJCV*, 1988, pp. 333–356.

[2] Bajcsy, R., "Active Perception," in *Proc. IEEE*, August 1988, pp. 996–1005.

[3] Ballard, D. H., "Animate Vision," in *AI*, vol. 48, no. 1, Feb. 1991, pp. 57–86.

[4] Tron, R. and Vidal, R., "Distributed Computer Vision Algorithms," in *IEEE Signal Processing Magazine*, May 2011, pp. 32–45.

[5] Macready, W. and Wolpert, D., "Distributed Constrained Optimization," in *International Conference on Complex Systems*, vol. Bar-Yam (Ed.). Perseus books, 2004.

[6] Soto, C. and Song, B. and Roy-Chowdhury, A. K., "Distributed multi-target tracking in a self-configuring camera network," in *Computer Vision and Pattern Recognition*, Aug. 2009, pp. 1486–1493. [Online]. Available: http://dx.doi.org/10.1109/CVPRW.2009.5206773

[7] Olfati-Saber, R., "Kalman-Consensus Filter: Optimality, Stability, and Performance," in *2009 Joint 48th IEEE-CDC and 28th Chinese Control Conf.*, Dec. 2009, pp. 7036–7042.

[8] Morye, A. A. and Ding, C. and Song, B. and Roy-Chowdhury, A. K. and Farrell, J. A., "Optimized Imaging and Target Tracking within a Distributed Camera Network," in *ACC*, Jun. 2011, pp. 474–480.

[9] Li, N. and Marden, J. R., "Designing Games for Distributed Optimization," in *CDC-ECE*. IEEE Transactions on Automatic Control, 2011, pp. 2434–2440.

[10] Zhu, M. and Martinez, S., "On Distributed Optimization under Inequality Constraints via Lagrangian Primal-Dual Methods," in *American Control Conference*, Jul. 2010, pp. 4863–4868.

[11] Song, B. and Soto, C. and Roy-Chowdhury, A. K. and Farrell, J. A., "Decentralized camera network control using game theory," in *Workshop on Smart Camera and Visual Sensor Networks at ICDSC*, Sep. 2008, pp. 1–8.

[12] Song, B. and Roy-Chowdhury, A. K., "Stochastic Adaptive Tracking in a Camera Network," in *ICCV*, Oct. 2007, pp. 1–8.

[13] Olfati-Saber, R. and Sandell, N. F., "Distributed Tracking in Sensor Networks with Limited Sensing Range," in *Proceedings of the American Control Conference*, Jun. 2008, pp. 3157–3162.

[14] Boyd, S. and Vandenberghe, L., "Convex Optimization." Cambridge University Press, 2004, pp. 241–246.

[15] Monderer, D. and Shapley, L. S., "Potential Games," in *Games and Economic Behavior*, vol. 14, no. 0044, 1996, pp. 124–143.

[16] Farrell, J. A., "Aided Navigation: GPS with High Rate Sensors." Mcgraw Hill, 2008, pp. 72–75.

[17] Trucco, E. and Verri, A., "Introductory Techniques for 3-D Computer Vision." Prentice Hall, 1998, pp. 26–27.

[18] Song, B. and Ding, C. and Kamal, A. T. and Farrell, J. A. and Roy-Chowdhury, A. K., "Distributed Camera Networks," in *IEEE Signal Processing Magazine*, May. 2011, pp. 20–31.

[19] Nedic, A. and Ozdaglar, A. and Parrilo, P. A., "Constrained Consensus and Optimization in Multi-Agent Networks," in *IEEE Transactions on Automatic Control*, vol. 55, Apr. 2010, pp. 922 – 938.

[20] Boyd, S. and Vandenberghe, L., "Convex Optimization." Cambridge University Press, 2004, pp. 397–409.

[21] ——, "Convex Optimization." Cambridge University Press, 2004, pp. 609–615.